SIMATIC
S7-1200 使用手册



SIMATIC S7-1200

Answers for industry.

SIEMENS

SIEMENS

SIMATIC

S7-1200 入门手册

设备手册

11/2009

A5E02486780-01

法律资讯

警告提示系统

为了您的人身安全以及避免财产损失，必须注意本手册中的提示。人身安全的提示用一个警告三角表示，仅与财产损失有关的提示不带警告三角。警告提示根据危险等级由高到低如下表示。

⚠危险
表示如果不采取相应的小心措施，**将会**导致死亡或者严重的人身伤害。

⚠警告
表示如果不采取相应的小心措施，**可能**导致死亡或者严重的人身伤害。

⚠小心
带有警告三角，表示如果不采取相应的小心措施，可能导致轻微的人身伤害。

小心
不带警告三角，表示如果不采取相应的小心措施，可能导致财产损失。

注意
表示如果不注意相应的提示，可能会出现不希望的结果或状态。

当出现多个危险等级的情况下，每次总是使用最高等级的警告提示。如果在某个警告提示中带有警告可能导致人身伤害的警告三角，则可能在该警告提示中另外还附带有可能导致财产损失的警告。

合格的专业人员

本文件所属的产品/系统只允许由符合各项工作要求的**合格人员**进行操作。其操作必须遵照各自附带的文件说明，特别是其中的安全及警告提示。 由于具备相关培训及经验，合格人员可以察觉本产品/系统的风险，并避免可能的危险。

按规定使用 Siemens 产品

请注意下列说明：

⚠警告
Siemens 产品只允许用于目录和相关技术文件中规定的使用情况。如果要使用其他公司的产品和组件，必须得到 Siemens 推荐和允许。正确的运输、储存、组装、装配、安装、调试、操作和维护是产品安全、正常运行的前提。必须保证允许的环境条件。必须注意相关文件中的提示。

商标

所有带有标记符号 ® 的都是西门子股份有限公司的注册商标。标签中的其他符号可能是一些其他商标，这是出于保护所有者权利的 目地由第三方使用而特别标示的。

责任免除

我们已对印刷品中所述内容与硬件和软件的一致性作过检查。然而不排除存在偏差的可能性，因此我们不保证印刷品中所述内容与硬件和软件完全一致。印刷品中的数据都按规定经过检测，必要的修正值包含在下一版本中。

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
德国

文件订购号: 6ES7298-8FA30-8KQ0
℗ 10/2009

前言

欢迎使用西门子 SIMATIC 控制器系列的最新产品 - S7-1200。 SIMATIC S7-1200 紧凑型控制器是一款节省空间的模块化控制器，适合要求简单或高级逻辑、HMI 和网络功能的小型自动化系统。 S7-1200 设计紧凑、成本低廉且功能强大，是控制小型应用的完美解决方案。

作为 SIMATIC“全集成自动化”(TIA, Totally Integrated Automation) 计划的一部分，S7-1200 产品系列和 STEP 7 Basic 编程工具为您提供了满足您自动化需求所需的灵活性。

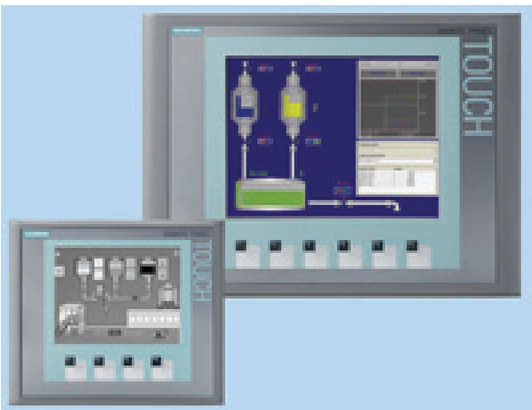
S7-1200 将帮助您轻松完成最具挑战性的任务！

SIMATIC S7-1200 控制器解决方案是专为“紧凑型”控制器类别设计的，由 SIMATIC S7-1200 控制器和 SIMATIC HMI 基本型面板组成，此二者均可使用 SIMATIC STEP 7 Basic 工程软件进行编程。 由于实现了使用同一个工程软件对两种设备进行编程，开发成本得以显著降低。



S7-1200 紧凑型控制器包括：

- 内置 PROFINET
- 能进行运动控制的高速 I/O、板载模拟量输入（将空间要求和对附加 I/O 的需求降到最低）、2 个用于 脉冲宽度应用 (页 83)的脉冲发生器以及最多 6 个 高速计数器 (页 78)
- CPU 模块中内置的板载 I/O 点提供 6 到 14 个输入点和 4 到 10 个输出点



用于 DC、继电器或模拟 I/O 的信号模块扩展了 I/O 点数，创新的信号板卡在 CPU 的正面 以提供附加 I/O (页 9)。

SIMATIC HMI 基本型面板 (页 10)是专为 S7-1200 设计的。

本入门手册介绍了 S7-1200 PLC。 下文概述了该设备的多个特征和功能。

更多相关信息，请参考*《S7-1200 可编程控制器系统手册》*。 也可使用以下网站搜索具体的产品信息，或与技术支持代表联系：

http://www.siemens.com/automation/support-request

有关 UL 和 FM 证书、CE 标签、C-Tick 和其它标准的信息，请参考 技术规范 (页 97)。

如需了解任何技术问题、培训或订购 S7 产品方面的信息，请与西门子经销商或销售部联系。西门子销售代表接受过技术培训并掌握有关操作、过程和工业以及您使用的各种西门子产品的最专业的知识，所以他们能够最快最高效地回答您可能遇到的任何问题。

目录

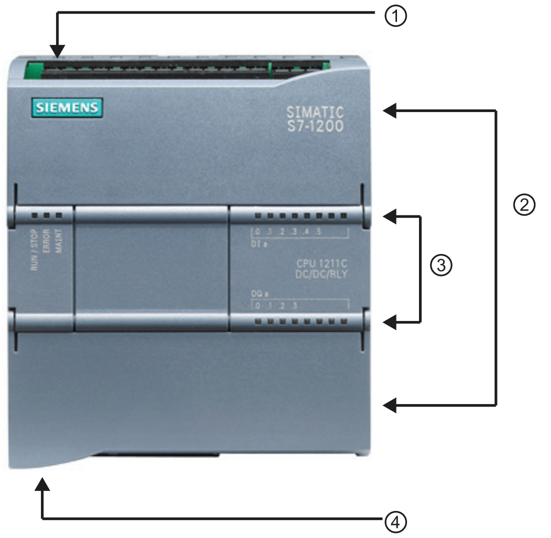
1 强大灵活的 S7-1200 简介

S7-1200 控制器使用灵活、功能强大，可用于控制各种各样的设备以满足您的自动化需求。 S7-1200 设计紧凑、组态灵活且具有功能强大的指令集，这些优势的组合使它成为控制各种应用的完美解决方案。

CPU 将微处理器、集成电源、输入和输出电路、内置 PROFINET、高速运动控制 I/O 以及板载模拟量输入组合到一个设计紧凑的外壳中以形成功能强大的控制器。 在您下载用户程序后，CPU 将包含监控应用中的设备所需的逻辑。 CPU 根据用户程序逻辑监视输入与更改输出，用户程序逻辑可以包含布尔逻辑、计数、定时、复杂数学运算以及与其它智能设备的通信。

为了与编程设备通信，CPU 提供了一个内置 PROFINET 端口。 借助 PROFINET 网络，CPU 可以与 HMI 面板或其它 CPU 通信。

为了确保应用程序安全，每个 S7-1200 CPU 都提供密码保护功能，用户通过它可以组态对 CPU 功能的访问。



① 电源接口
② 可拆卸用户接线连接器（保护盖下面）
③ 板载 I/O 的状态 LED
④ PROFINET 连接器（CPU 的底部）

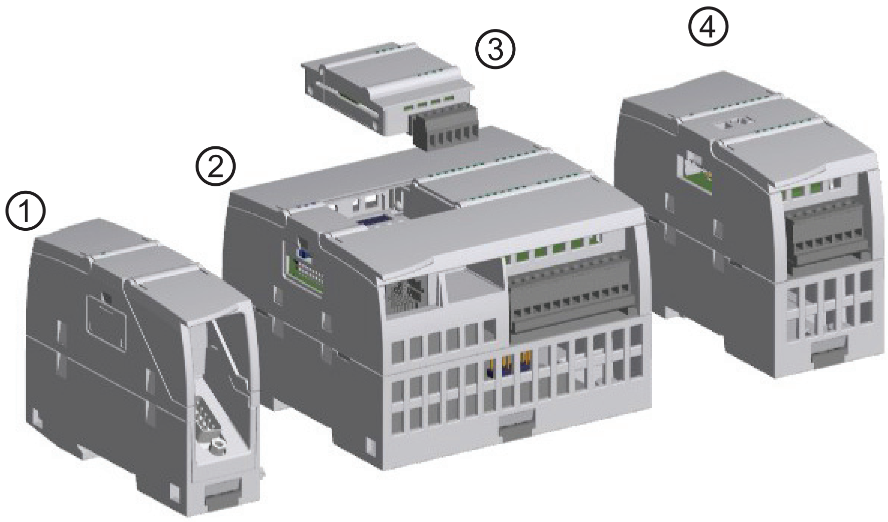
特征	CPU 1211C	CPU 1212C	CPU 1214C
物理尺寸 (mm)	90 x 100 x 75	90 x 100 x 75	110 x 100 x 75
用户存储器 • 工作存储器 • 装载存储器 • 保持存储器	 • 25 KB • 1 MB • 2 KB	 • 25 KB • 1 MB • 2 KB	 • 50 KB • 2 MB • 2 KB
本地板载 I/O • 数字量 • 模拟量	 • 6 点输入 4 点输出 • 2 路输入	 • 8 点输入 6 点输出 • 2 路输入	 • 14 点输入 10 点输出 • 2 路输入
过程映像大小 • 输入 • 输出	 • 1024 个字节 • 1024 个字节	 • 1024 个字节 • 1024 个字节	 • 1024 个字节 • 1024 个字节
位存储器 (M)	4096 个字节	4096 个字节	8192 个字节
信号模块扩展	无	2	8
信号板	1	1	1
通信模块	3	3	3
高速计数器 • 单相 • 正交相位	3 • 3 个，100 kHz • 3 个，80 kHz	4 • 3 个，100 kHz 1 个，30 kHz • 3 个，80 kHz 1 个，20 kHz	6 • 3 个，100 kHz 3 个，30 kHz • 3 个，80 kHz 3 个，20 kHz
脉冲输出 [1]	2	2	2
存储卡（选件）	有	有	有
实时时钟保持时间	通常为 10 天/40 摄氏度时最少 6 天		
实数数学运算执行速度	18 μs/指令		
布尔运算执行速度	0.1 μs/指令		

[1] 仅 DC 输出型（非继电器型）CPU 支持脉冲输出。

不同的 CPU 型号提供了各种各样的特征和功能，这些特征和功能可帮助用户针对不同的应用创建有效的解决方案。有关特定 CPU 的详细信息，请参见 技术规范 (页 102)。

1.1 扩展 CPU 的能力

S7-1200 系列提供了各种信号模块和信号板用于扩展 CPU 的能力。 还可以安装附加的通信模块以支持其它通信协议。 有关特定模块的详细信息，请参见 技术规范 (页 97)。

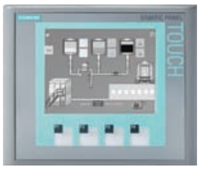


① 通信模块 (CM)	③ 信号板 (SB)
② CPU	④ 信号模块 (SM)

模块		仅输入	仅输出	输入/输出组合
信号模块 (SM)	数字量	8 x DC 输入	8 x DC 输出 8 x 继电器输出	8 x DC 输入/8 x DC 输出 8 x DC 输入/8 x 继电器输出
		16 x DC 输入	16 x DC 输出 16 x 继电器输出	16 x DC 输入/16 x DC 输出 16 x DC 输入/16 x 继电器输出
	模拟量	4 x 模拟量输入 8 x 模拟量输入	2 x 模拟量输出 4 x 模拟量输出	4 x 模拟量输入/2 x 模拟量输出
信号板 (SB)	数字量	-	-	2 x DC 输入/2 x DC 输出
	模拟量	-	1 x 模拟量输出	-
通信模块 (CM) • RS485 • RS232				

1.2 HMI 基本型面板

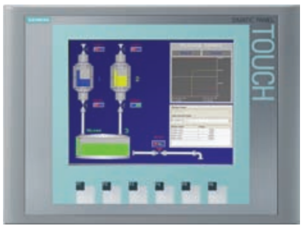
由于可视化已成为大多数机器设计的标准组件，所以 SIMATIC HMI 基本型面板提供了用于执行基本操作员监控任务的触摸屏设备。 所有面板的保护等级均为 IP65 并通过 CE、UL、cULus 和 NEMA 4x 认证。



KTP 400 Basic PN

- 单色（STN，灰度）
- 4" 触摸屏，带 4 个触摸键
- 纵向或横向
- 尺寸： 3.8"
- 分辨率： 320 x 240

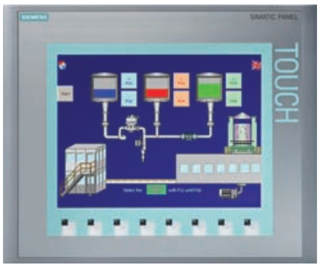
- 128 个变量
- 50 个过程画面
- 200 个报警
- 25 条曲线
- 32 KB 配方存储器
- 5 个配方，20 条数据记录，20 个条目



KTP 600 Basic PN

- 彩色（TFT，256 色）或单色（STN，灰度）
- 6" 触摸屏，带 6 个触摸键
- 纵向或横向
- 尺寸：5.7"
- 分辨率： 320 x 240

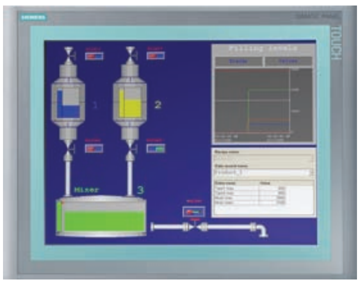
- 128 个变量
- 50 个过程画面
- 200 个报警
- 25 条曲线
- 32 KB 配方存储器
- 5 个配方，20 条数据记录，20 个条目



KTP1000 Basic PN

- 彩色（TFT，256 色）
- 10" 触摸屏，
  带 8 个触摸键
- 尺寸：10.4"
- 分辨率： 640 x 480

- 256 个变量
- 50 个过程画面
- 200 个报警
- 25 条曲线
- 32 KB 配方存储器
- 5 个配方，20 条数据记录，20 个条目



TP1500 Basic PN

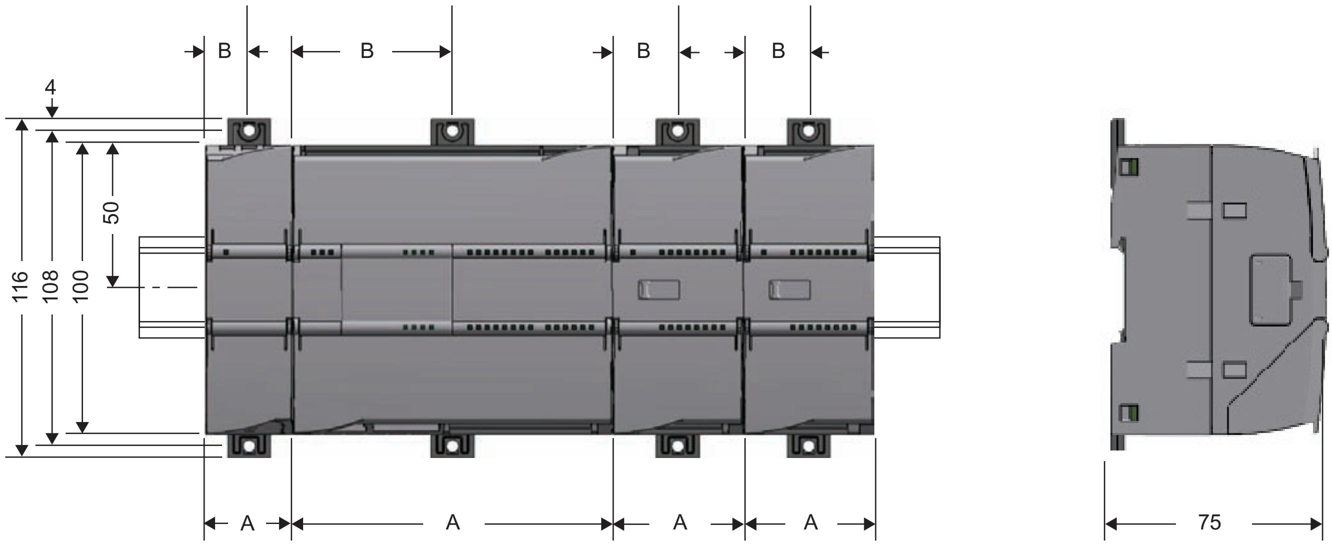
- 彩色（TFT，256 色）
- 15" 触摸屏
- 尺寸：15.1"
- 分辨率： 1024 x 768

- 256 个变量
- 50 个过程画面
- 200 个报警
- 25 条曲线
- 32 KB 配方存储器（集成闪存）
- 5 个配方，20 条数据记录，20 个条目

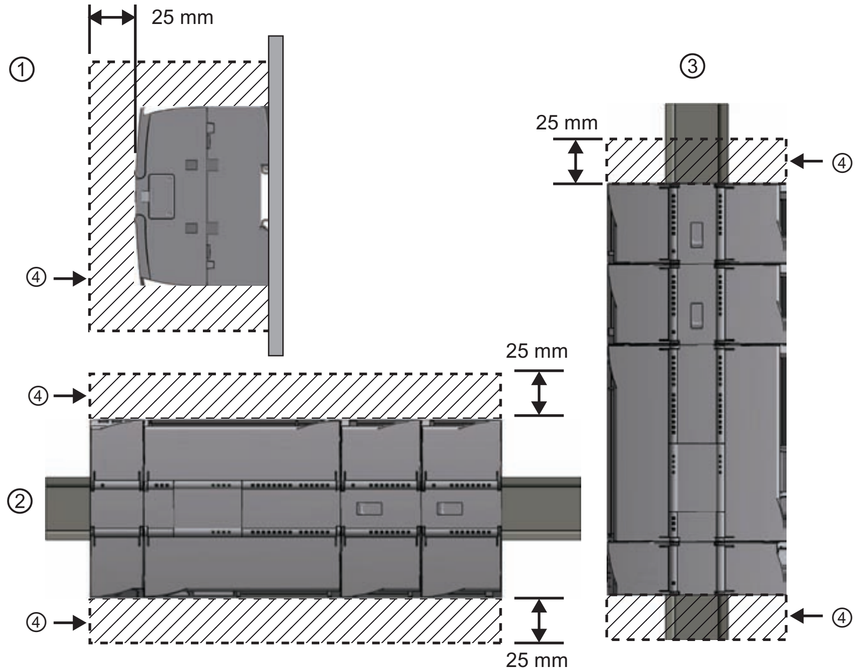
1.3 安装尺寸和间隙要求

S7-1200 PLC 设计得易于安装。 无论安装在面板上还是标准 DIN 导轨上，其紧凑型设计都有利于有效利用空间。

CPU、SM 和 CM 支持 DIN 导轨安装和面板安装。 使用模块上的 DIN 导轨卡夹将设备固定到导轨上。 这些卡夹还能掰到一个伸出位置以提供将设备直接安装到面板上的螺钉安装位置。 设备上 DIN 卡夹的孔内部尺寸是 4.3 mm。



S7-1200 设备		宽度 A	宽度 B
CPU	CPU 1211C 和 CPU 1212C	90 mm	45 mm
	CPU 1214C	110 mm	55 mm
信号模块 (SM)	8 和 16 点 DC 和继电器型（8I、16I、8Q、16Q、8I/8Q） 模拟量（4AI、8AI、4AI/4AQ、2AQ、4AQ）	45 mm	22.5 mm
	16I/16Q 继电器型 (16I/16Q)	70 mm	35 mm
通信模块 (CM)	CM 1241 RS232 和 CM 1241 RS485	30 mm	15 mm



①	侧视图	③	垂直安装
②	水平安装	④	空隙区域



规划安装时，务必注意以下指导原则：

- 将设备与热辐射、高压和电噪声隔离开。
- 留出足够的空隙以便冷却和接线。 必须在设备的上方和下方留出 25 mm 的发热区以便空气自由流通。

有关安装的具体要求和指导原则，请参考*《S7-1200 系统手册》*。

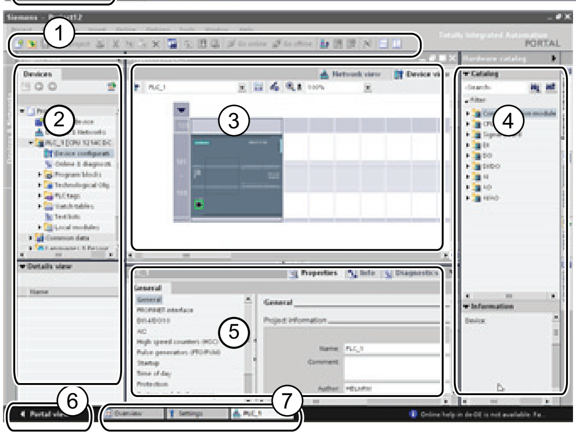
STEP 7 Basic 使工作更轻松 2

STEP 7 Basic 提供了一个用户友好的环境，供用户开发控制器逻辑、组态 HMI 可视化和设置网络通信。 为帮助用户提高生产率，STEP 7 Basic 提供了两种不同的项目视图： 根据工具功能组织的面向任务的门户集（门户视图），或项目中各元素组成的面向项目的视图（项目视图）。 请选择能让您的工作最高效的视图。只需通过单击就可以切换门户视图和项目视图。



门户视图提供项目任务的功能视图，并根据要完成的任务来组织工具。 用户可以很容易地确定如何继续以及选择哪个任务。

① 不同任务的门户
② 所选门户的任务
③ 所选操作的选择面板
④ 切换到项目视图



项目视图提供了访问项目中任意组件的途径。

① 菜单和工具栏
② 项目浏览器
③ 工作区
④ 任务卡
⑤ 巡视窗口
⑥ 切换到门户视图
⑦ 编辑器栏

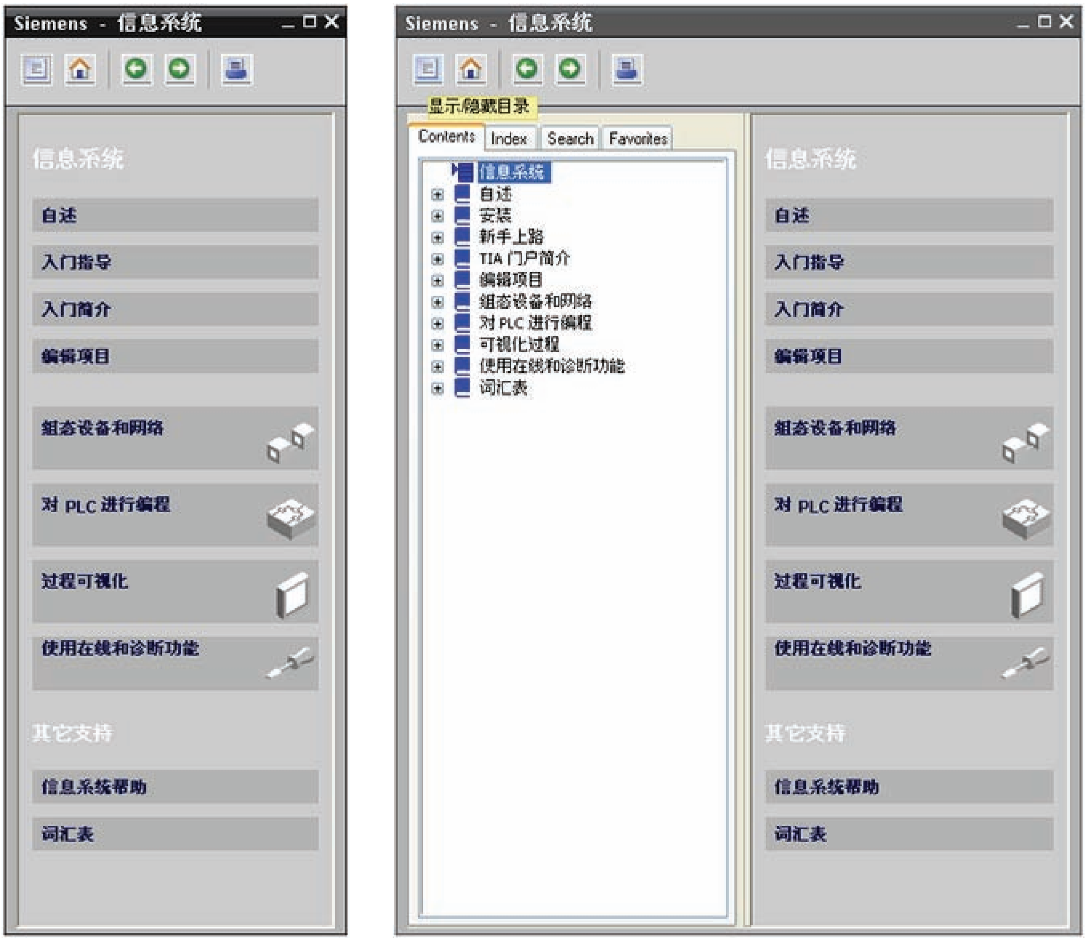
由于这些组件组织在一个视图中，所以您可以方便地访问项目的各个方面。 例如，巡视窗口显示了用户在工作区中所选对象的属性和信息。 当用户选择不同的对象时，巡视窗口会显示用户可组态的属性。 巡视窗口包含用户可用于查看诊断信息和其它消息的选项卡。

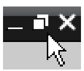
编辑器栏会显示所有打开的编辑器，从而帮助用户更快速和高效地工作。 要在打开的编辑器之间切换，只需单击不同的编辑器。 还可以将两个编辑器垂直或水平排列在一起显示。 通过该功能可以在编辑器之间进行拖放操作。

2.1 在您需要时提供的帮助

为帮助用户获得更多信息或快速而高效地解决问题，STEP 7 Basic 提供了智能的需求点帮助。 例如，界面中的某些工具提示（例如，指令的工具提示）通过“层叠”提供更多信息。 工具提示旁的黑色三角形表示有更多信息。

STEP 7 Basic 提供了丰富全面的在线信息和帮助系统，该系统介绍了用户已安装的所有 SIMATIC TIA 产品。 该信息系统会在一个不会遮挡工作区域的窗口中打开。 单击信息系统中的“显示/隐藏目录”(Show/hide contents) 按钮可显示目录和移除帮助窗口。 随后可以调整帮助窗口的大小。





如果 STEP 7 Basic 已最大化，则单击“显示/隐藏目录”按钮将不会移除帮助窗口。 单击 STEP 7 Basic 中的“向下还原”按钮可移除帮助窗口。 随后可以移动和调整帮助窗口的大小。

2.1.1 打印在线帮助的主题



要从信息系统中打印，请单击帮助窗口中的“打印”(Print) 按钮。



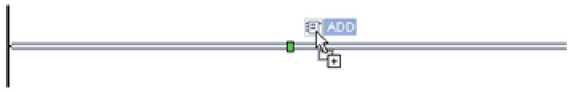
通过“打印”(Print) 对话框可以选择要打印的主题。 确保面板显示了主题。 然后可以选择任何要打印的其它主题。 单击“打印”(Print) 按钮将所选主题发送到打印机。

2.2 提供易于使用的工具

2.2.1 轻松向用户程序中插入指令

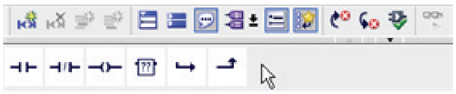
STEP 7 Basic 提供了包含各种程序指令的任务卡。 这些指令按功能分组。

要创建程序，可将指令从任务卡拖动到程序段中。





2.2.2 从工具栏轻松访问收藏的指令



STEP 7 Basic 提供了“收藏夹”(Favorites) 工具栏，用户可通过该工具栏快速访问常用的指令。

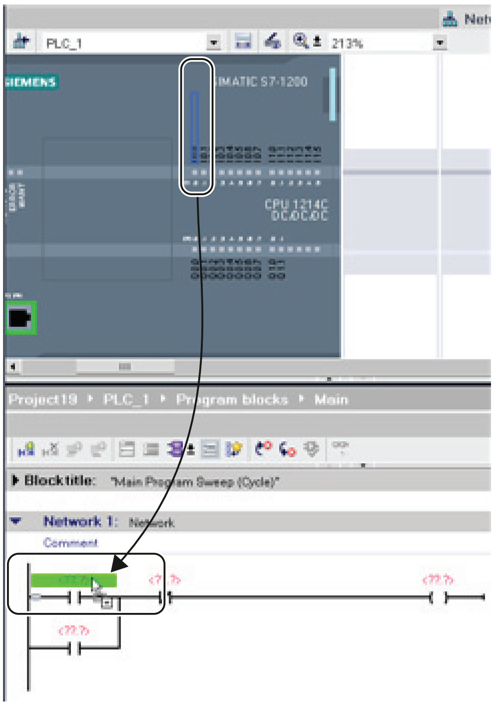
只需单击指令的图标即可将其插入程序段！

用户可以通过添加新指令方便地自定义“收藏夹”(Favorites)。 只需将指令拖放到“收藏夹”(Favorites)。 如此单击即可插入该指令！





2.2.3 在编辑器之间轻松拖放



为帮助用户快速和方便地执行任务，STEP 7 Basic 允许用户将元素从一个编辑器拖放到另一个编辑器中。 例如，可以将 CPU 的输入拖动到用户程序中指令的地址上。（必须放大至少 200% 才能选中 CPU 的 I/O。）

请注意，变量名称不仅会在 PLC 变量表中显示，还会在 CPU 上显示。

要一次显示两个编辑器，请使用“拆分编辑器”(Split editor) 菜单命令或工具栏中的相应按钮。





要在已打开的编辑器之间切换，请单击编辑器栏中的图标。



2.2.4 轻松更改 CPU 的工作模式

该 CPU 没有用于更改工作模式（STOP 或 RUN）的物理开关。 在设备配置中组态 CPU 时，应组态 CPU 属性中的启动行为 (页 44)。 “在线和诊断”(Online and Diagnostics) 门户提供了用于更改在线 CPU 工作模式的操作员面板。

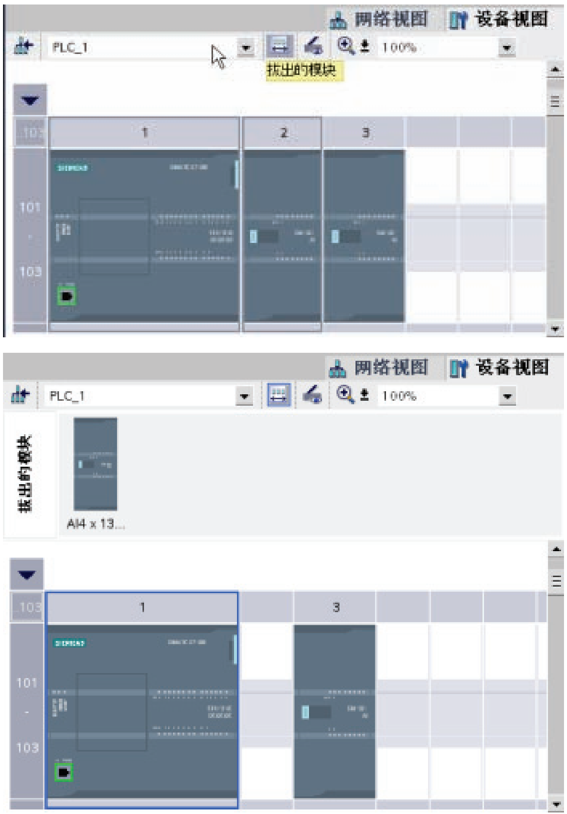
要使用 CPU 操作员面板，必须在线连接到 CPU。 “在线工具”(Online tools) 任务卡显示的操作员面板显示了在线 CPU 的工作模式。 也可以通过该操作员面板更改在线 CPU 的工作模式。



使用操作员面板上的按钮更改工作模式（STOP 或 RUN）。 操作员面板还提供了用于复位存储器的 MRES 按钮。

RUN/STOP 指示器的颜色指示 CPU 当前的工作模式。 黄色表示 STOP 模式，而绿色表示 RUN 模式。

2.2.5 轻松实现实际“拔出”模块而不会丢失组态数据



STEP 7 Basic 为“拔出的”模块提供了一个存储区域。 用户可以从机架中拖出模块以保存该模块的组态。 这些拔出的模块会随项目一同保存，从而在将来不必重新组态参数即可再次插入相应模块。

此功能的其中一种用途是用于临时维护。 想想用户可能正等待一个替换模块，并计划临时使用一个不同的模块来短期替换相应模块。 此时，用户可以将组态的模块从机架拖动到“拔出的模块”(Unplugged modules) 区域，然后插入临时模块。

只要该模块具有相同的基本编址数据，替换模块操作就不会影响 PLC 变量。 例如，可以使用一个 8X8 组合的数字 SM 或一个 16 点输入的数字 SM 来替换一个 8 点输入的数字 SM。

2.2.6 轻松修改 STEP 7 Basic 的外观和组态



用户可以选择各种各样的设置，例如界面的外观、语言或保存工作的目录。

在“选项”(Options) 菜单中选择“设置”(Settings) 命令更改这些设置。

入门指南

3

使用 STEP 7 Basic 非常容易！ 在接下来的几页中，您可以看到开始创建项目有多么快捷。



在“开始”(Start) 门户中，单击“创建新项目”(Create new project) 任务。

输入项目名称并单击“创建”(Create) 按钮。



创建项目后，选择“设备和网络”(Devices & Networks) 门户。

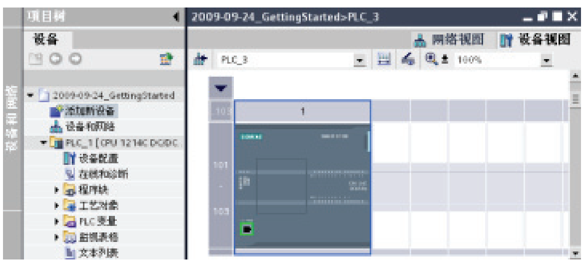
单击“添加新设备”(Add new device) 任务。



选择要添加到项目中的 CPU：

1. 在“添加新设备”(Add new device) 对话框中，单击“SIMATIC PLC”按钮。
2. 从列表中选择一个 CPU。
3. 单击“添加”(Add) 按钮，将所选 CPU 添加到项目中

请注意，“打开设备视图”(Open device view) 选项已被选中。 在该选项被选中的情况下单击“添加”(Add) 将打开项目视图的“设备配置”(Device configuration)。

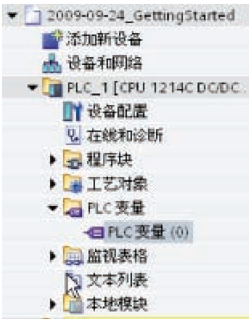


设备视图显示所添加的 CPU。

为 CPU 的 I/O 创建变量

说明

“PLC 变量”是 I/O 和地址的符号名称。 用户创建 PLC 变量后，STEP 7 Basic 会将变量存储在变量表中。 项目中的所有编辑器（例如程序编辑器、设备编辑器、可视化编辑器和监视表格编辑器）均可访问该变量表。



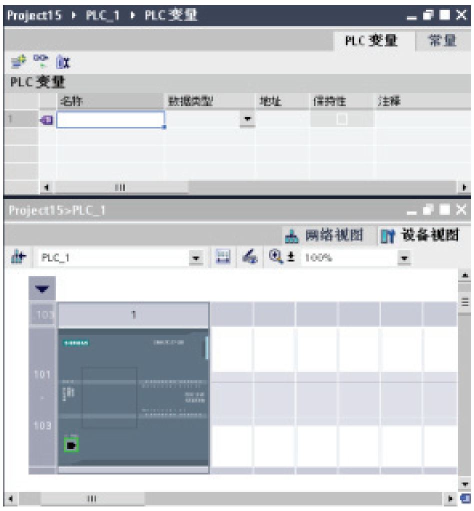
若设备编辑器已打开，用户就可打开变量表。

您可在在编辑器栏中看到已打开的编辑器。

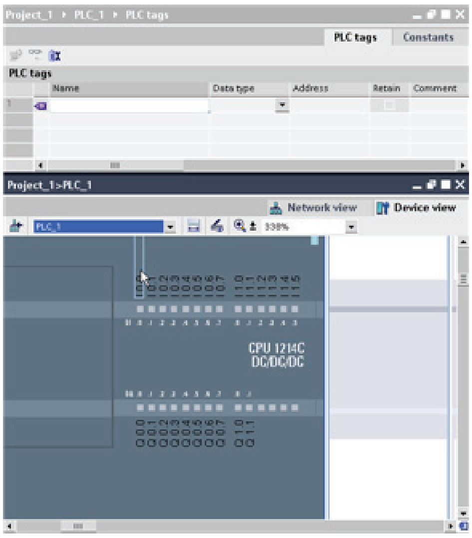




在工具栏中，单击“水平拆分编辑器空间”(Split editor space horizontally) 按钮。

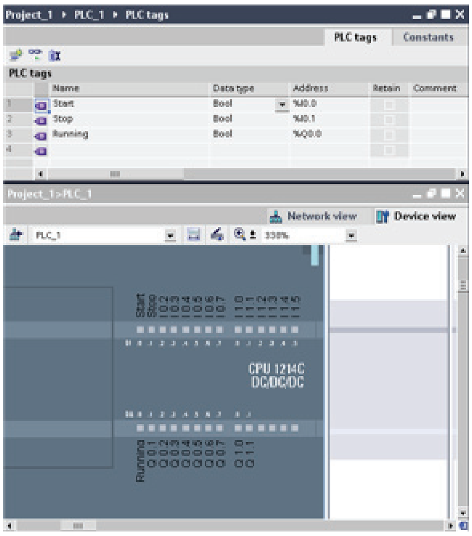


STEP 7 Basic 即会将变量表和设备编辑器显示在一起。



将设备配置放大 200% 以上，以便能清楚的查看并选择 CPU 的 I/O 点。

1. 选择 I0.0 并将其拖动到变量表的第一行。
2. 将变量名称从“I0.0”更改为“Start”。
3. 将 I0.1 拖动到变量表，并将名称更改为“Stop”。
4. 将 CPU 底部的 Q0.0 拖动到变量表，并将名称更改为“Running”。



将 PLC 变量输入变量表之后，在用户程序中即可使用这些变量。

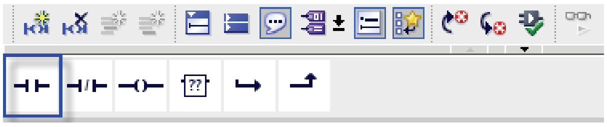
在用户程序中创建一个简单程序段

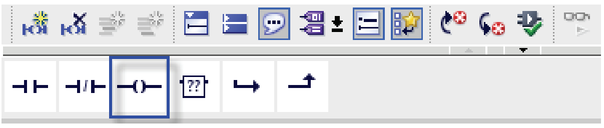


程序代码由 PLC 依次执行的指令组成。 在本实例中，使用梯形图 (LAD) 创建程序代码。 LAD 程序是一系列类似梯级的程序段。

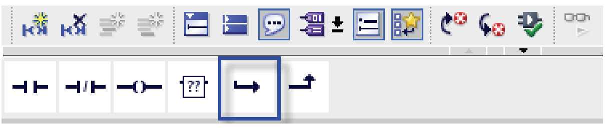
要打开程序编辑器，请按以下步骤操作：

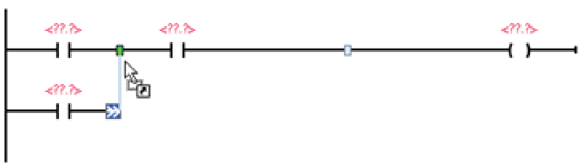
1. 在项目树中展开“程序块”(Program blocks) 文件夹以显示“Main [OB1]”块。
2. 双击“Main [OB1]”块。













程序编辑器将打开程序块 (OB1)。 使用“收藏夹”(Favorites) 上的按钮将触点和线圈插入程序段中：

1. 单击“收藏夹”(Favorites) 上的“常开触点”按钮向程序段添加一个触点。
2. 在本实例中，添加第二个触点。
3. 单击“输出线圈”(Output coil) 按钮插入一个线圈。

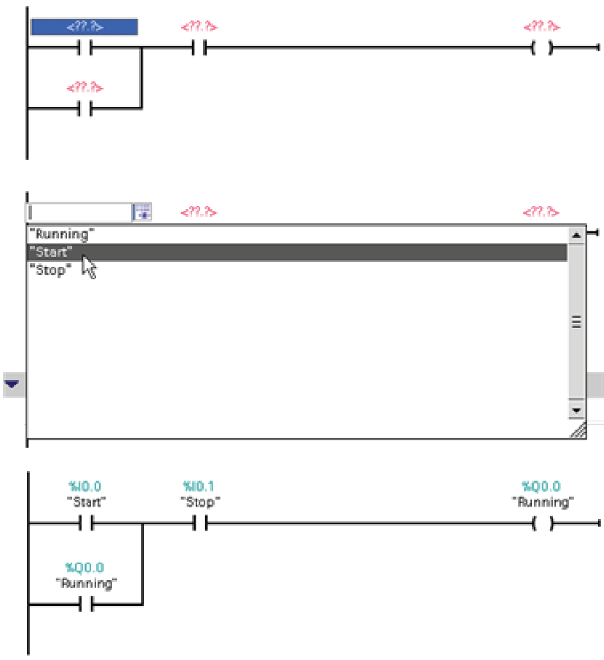
“收藏夹”(Favorites) 还提供了用于创建分支的按钮：

1. 单击“打开分支”(Open branch) 图标向程序段的电源线添加分支。
2. 在打开的分支中插入另一个常开触点。
3. 将双向箭头拖动到第一梯级上断开和闭合触点之间的一个连接点位置（梯级上的绿色方块）。

要保存项目，请单击工具栏中的“保存项目”(Save project) 按钮。 请注意，在保存前不必完成对梯级进行编辑。

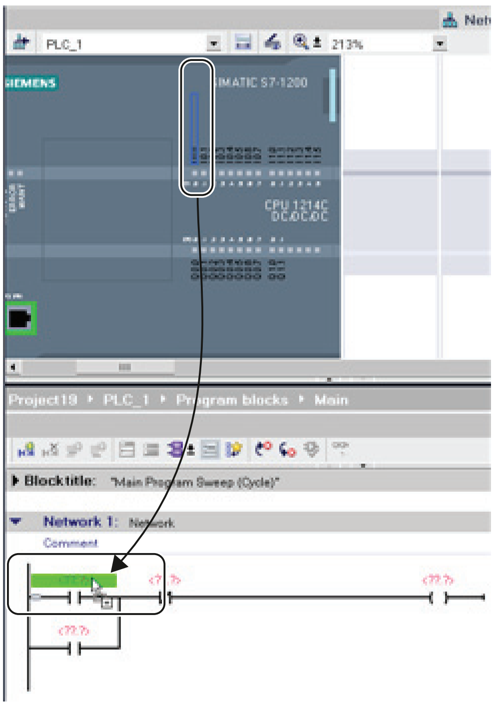
您已创建了一个 LAD 指令的程序段。 现在可以将变量名称与这些指令进行关联。

使用变量表中的 PLC 变量对指令进行寻址



使用变量表，用户可以快速输入对应触点和线圈地址的 PLC 变量。

1. 双击第一个常开触点上方的默认地址 *<??.?>*。
2. 单击地址右侧的选择器图标打开变量表中的变量。
3. 从下拉列表中，为第一个触点选择“Start”。
4. 对于第二个触点，重复上述步骤并选择变量“Stop”。
5. 对于线圈和锁存触点，选择变量“Running”。



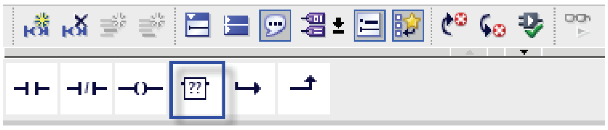
还可以直接从 CPU 中拖拽 I/O 地址。 为此，只需 拆分项目视图的工作区 (页 17)。

必须将 CPU 放大 200% 以上才能选择 I/O 点。

可以将“设备配置”(Device configuration) 中 CPU 上的 I/O 拖到程序编辑器的 LAD 指令上，这样不仅会创建指令的地址，还会在 PLC 变量表中创建相应条目。

向第二个程序段添加数学运算指令

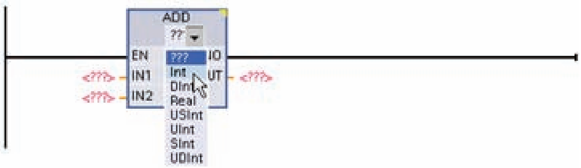
程序编辑器提供了一个通用“功能框”指令。 插入此功能框指令之后，可从下拉列表中选择指令类型，例如 ADD 指令。



单击“收藏夹”(Favorites) 工具栏中的通用“功能框”指令。

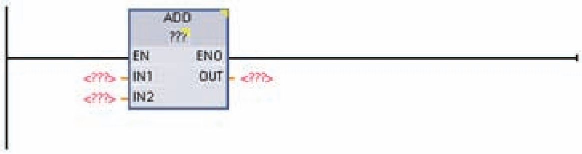




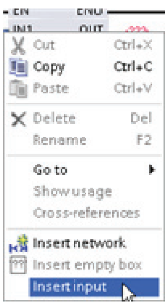


通用“功能框”指令支持多种指令。 在本实例中，创建一个 ADD 指令：

1. 单击功能框指令黄色角以显示指令的下拉列表。
2. 向下滚动列表并选择 ADD 指令。
3. 单击“?”旁边的黄色角为输入和输出选择数据类型。



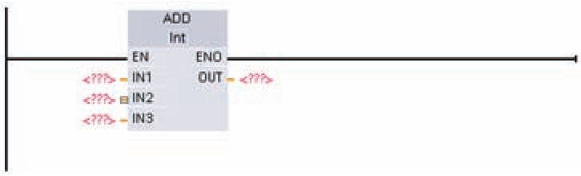
现在即可为 ADD 指令所用的值输入变量（或存储器地址）。



还可以为某些指令创建更多输入：

1. 单击一个输入。
2. 单击右键以显示快捷菜单并选择“插入输入”(Insert input) 命令。

ADD 指令现在即使用三个输入。



向项目中添加 HMI 设备

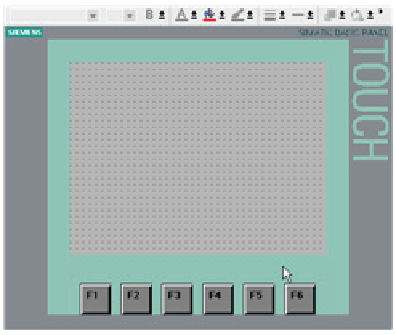


向项目中添加 HMI 设备非常容易！



1. 双击“添加新设备”(Add new device) 图标。
2. 在“添加新设备”(Add new device) 对话框中单击“SIMATIC HMI”按钮。
3. 从列表中选择特定的 HMI 设备。

   可以运行 HMI 向导来组态 HMI 设备的画面。
4. 单击“确定”(OK) 将 HMI 设备添加到项目中。

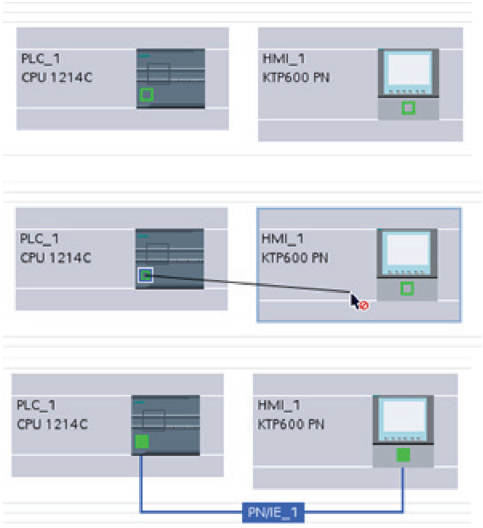


HMI 设备即添加到项目中。

STEP 7 Basic 提供了一个 HMI 向导，可以帮助用户组态 HMI 设备的所有画面和结构。

如果未运行 HMI 向导，则 STEP 7 Basic 将创建一个简单的默认 HMI 画面。

在 CPU 和 HMI 设备之间创建网络连接



创建网络非常简单！

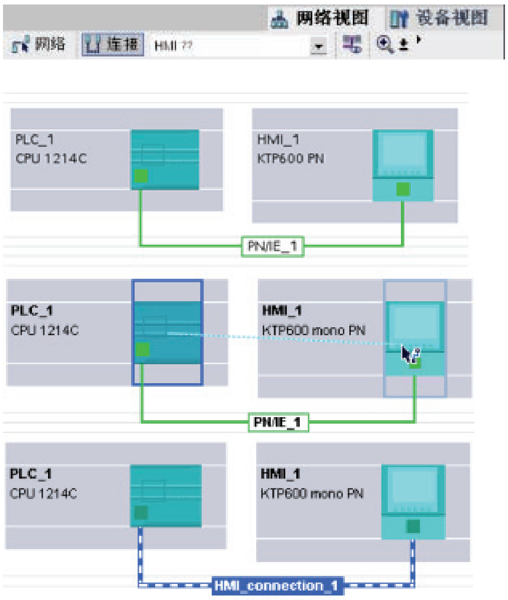
转到“设备和网络”(Devices and Networks) 并选择网络视图来显示 CPU 和 HMI 设备。

要创建 PROFINET 网络，只需从一个设备的绿色框拖出一条线连接到另一个设备的绿色框（以太网端口）。

随即会为这两个设备创建一个网络连接。

创建 HMI 连接以共享变量

通过在两个设备之间创建 HMI 连接，用户可以轻松地在两个设备之间共享变量。



选择相应的网络连接，单击“HMI 连接”(HMI connection) 按钮。

HMI 连接会将相关的两个设备变为蓝色。

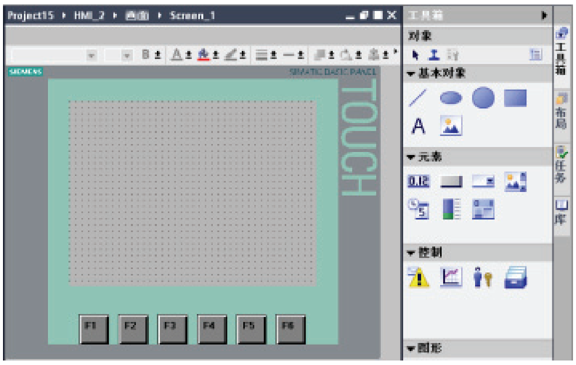
选择 CPU 设备并拖出一条线连接到 HMI 设备。

该 HMI 连接允许用户通过选择 PLC 变量列表对 HMI 变量进行组态。

用户可以采用其它方法创建 HMI 连接：

- 通过从 PLC 变量表、程序编辑器或设备配置编辑器将 PLC 变量拖动 HMI 画面编辑器，自动创建 HMI 连接。
- 通过使用 HMI 向导浏览到相应 PLC，自动创建 HMI 连接。

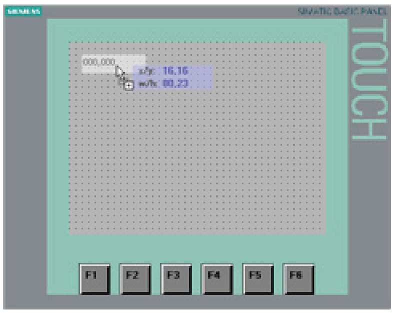
创建 HMI 画面



即使不利用 HMI 向导，组态 HMI 画面也很容易。

STEP 7 Basic 提供了一个标准库集合，用于插入基本形状、交互元素，甚至是标准图形。

要添加元素，只需将其中一个元素拖放到画面中。





使用元素的属性（在巡视窗口中）组态该元素的外观和特性。

还可以通过从项目树或程序编辑器将 PLC 变量拖放到 HMI 画面来创建画面上的元素。PLC 变量即成为画面上的元素 然后可以使用属性来更改该元素的参数。

为 HMI 元素选择 PLC 变量

在画面上创建元素后，使用所选元素的属性将 PLC 变量分配给该元素。 单击“连接”(Connections) 域中的按钮将显示 CPU 的 PLC 变量。



也可以从项目树将 PLC 变量拖放到 HMI 画面中。 在项目树的“详细信息”视图中显示 PLC 变量，然后将其拖放到 HMI 画面中。

简化了 PLC 概念 4

4.1 每个扫描周期均执行的任务

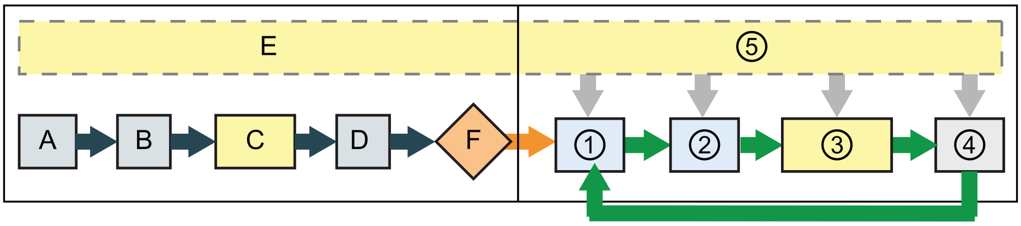
每个扫描周期都包括写入输出、读取输入、执行用户程序指令以及执行系统维护或后台处理。 该周期称为扫描周期或扫描。 在默认条件下，所有数字和模拟 I/O 点都通过内部存储区（即过程映像）与扫描周期进行同步更新。 过程映像包含 CPU、信号板和信号模块上的物理输入和输出的快照。



CPU 仅在用户程序执行前读取物理输入，并将输入值存储在过程映像输入区。 这样可确保这些值在整个用户指令执行过程中保持一致。

CPU 执行用户指令逻辑，并更新过程映像输出区中的输出值，而不是写入实际的物理输出。

执行完用户程序后，CPU 将所生成的输出从过程映像输出区写入到物理输出。



STARTUP		RUN	
A	清除输入（或“I”）存储器	①	将 Q 存储器写入物理输出
B	使用上一个值或替换值对输出执行初始化	②	将物理输入的状态复制到 I 存储器
C	执行启动 OB	③	执行程序循环 OB
D	将物理输入的状态复制到 I 存储器	④	执行自检诊断
E	将所有中断事件存储到要在 RUN 模式下处理的队列中	⑤	在扫描周期的任何阶段处理中断和通信
F	启用将输出（或“Q”）存储器的值写入到物理输出		

这一过程通过在给定周期内执行用户指令而提供一致的逻辑，并防止物理输出点可能在过程映像输出区中多次改变状态而出现抖动。

可通过将模块从 I/O 的自动更新中删除来更改其默认行为。也可在执行指令时立即读取数字和模拟 I/O 值并将其写入模块。 立即读取物理输入并不会更新过程映像输入区。 立即写入物理输出会同时更新过程映像输出区和物理输出点。

4.2 CPU 的工作模式

CPU 有以下三种工作模式： STOP 模式、STARTUP 模式和 RUN 模式。 CPU 前面的状态 LED 指示当前工作模式。

- 在 STOP 模式下，CPU 不执行任何程序，而用户可以下载项目。
- 在 STARTUP 模式下，CPU 会执行任何启动逻辑（如果存在）。 在 STARTUP 模式下不处理任何中断事件。
- 在 RUN 模式下，重复执行扫描周期。 在程序循环阶段的任何时刻都可能发生和处理中断事件。

说明

CPU 处于 RUN 模式下时，无法下载任何项目。 只有在 CPU 处于 STOP 模式时，才能下载项目。

CPU 支持通过暖启动方法进入 RUN 模式。 暖启动不包括存储器复位，但通过编程软件可以控制存储器复位。 存储器复位将清除所有工作存储器、保持性及非保持性存储区，并将装载存储器内容复制到工作存储器。 存储器复位不会清除诊断缓冲区，也不会清除永久保存的 IP 地址。 在暖启动时，所有非保持性系统及用户数据都将被初始化。

可以使用编程软件指定 CPU 的上电模式以及重启方法。 该组态项目出现在 CPU“设备配置”(Device Configuration) 的“启动”(Startup) 下。 通电后，CPU 将执行一系列上电诊断检查和系统初始化操作。 然后 CPU 进入适当的上电模式。 检测到的某些错误将阻止 CPU 进入 RUN 模式。 CPU 支持以下上电模式： STOP 模式、“暖启动后转到 RUN 模式”和“暖启动后转到上一个模式”。



CPU 未提供用于更改工作模式的物理开关。 请使用 STEP 7 Basic 在线工具中的 CPU 操作员面板更改工作模式（STOP 或 RUN）。

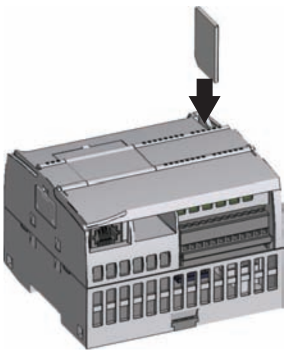
也可在程序中包含 STP 指令，以使 CPU 切换到 STOP 模式。 这样就可以根据程序逻辑停止程序的执行。

4.3 存储区、寻址和数据类型

CPU 提供了以下用于存储用户程序、数据和组态的存储区：

- 装载存储器用于非易失性地存储用户程序、数据和组态。 项目被下载到 CPU 后，首先存储在装载存储区中。 该存储区位于存储卡（如存在）或 CPU 中。 该非易失性存储区能够在断电后继续保持。存储卡支持的存储空间比 CPU 内置的存储空间更大。
- 工作存储器是易失性存储器，用于在执行用户程序时存储用户项目的某些内容。 CPU 会将一些项目内容从装载存储器复制到工作存储器中。该易失性存储区将在断电后丢失，而在恢复供电时由 CPU 恢复。
- 保持性存储器用于非易失性地存储限量的工作存储器值。 保持性存储区用于在断电时存储所选用户存储单元的值。 发生掉电时，CPU 留出了足够的缓冲时间来保存几个有限的指定单元的值。 这些保持性值会随后在上电时恢复。

可选的 SIMATIC 存储卡可用作存储用户程序的替代存储器，或用于传送程序。 如果使用存储卡，CPU 将运行存储卡中的程序而不是自身存储器中的程序。





CPU 仅支持预格式化的 SIMATIC 存储卡。

要插入存储卡，需打开 CPU 顶盖，然后将存储卡插入到插槽中。推弹式连接器可以轻松地插入和取出。存储卡要求正确安装。

检查以确定存储卡没有写保护。 滑动保护开关，使其离开“Lock”位置。

将可选的 SIMATIC 存储卡用作程序卡或传送卡：

- 使用传送卡可将项目复制到多个 CPU，而无需使用 STEP 7 Basic。 传送卡将所存储的项目从卡中复制到 CPU 的存储器。 将程序复制到 CPU 后必须取出传送卡。
- 程序卡可以替代 CPU 存储器；所有 CPU 功能都由该程序卡进行控制。 插入程序卡会擦除 CPU 内部装载存储器的所有内容（包括用户程序和任何强制 I/O）。 CPU 然后会执行程序卡中的用户程序。

  程序卡**必须**保留在 CPU 中。 如果取出程序卡，CPU 将切换到 STOP 模式。

S7-1200 支持的数据类型



数据类型用于指定数据元素的大小以及如何解释数据。 每个指令参数至少支持一种数据类型，而有些参数支持多种数据类型。 将光标停在指令的参数域上方，便可看到给定参数所支持的数据类型。

数据类型	大小（位）	范围	常量输入实例
Bool	1	0 到 1	TRUE, FALSE, 0, 1
Byte	8	16#00 到 16#FF	16#12, 16#AB
Word	16	16#0000 到 16#FFFF	16#ABCD, 16#0001
DWord	32	16#00000000 到 16#FFFFFFFF	16#02468ACE
Char	8	16#00 到 16#FF	'A', 't', '@'
Sint	8	-128 到 127	123, -123
Int	16	-32,768 到 32,767	123, -123
Dint	32	-2,147,483,648 到 2,147,483,647	123, -123
USInt	8	0 到 255	123
UInt	16	0 到 65,535	123
UDInt	32	0 到 4,294,967,295	123
Real	32	+/-1.18 x 10^{-38} 到 +/-3.40 x 10^{38}	123.456、-3.4、-1.2E+12、3.4E-3
LReal	64	+/-2.23 x 10^{-308} 到 +/-1.79 x 10^{308}	12345.123456789 -1.2E+40
Time	32	T#-24d_20h_31m_23s_648ms to T#24d_20h_31m_23s_647ms 存储形式： -2,147,483,648 ms to +2,147,483,647 ms	T#5m_30s 5#-2d T#1d_2h_15m_30x_45ms

数据类型	大小（位）	范围	常量输入实例
String	变量	0 到 254 字节字符	'ABC'
DTL[1]	12 个字节	最小值： DTL#1970-01-01-00:00:00.0 最大值： DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20:30:20.250

[1] DTL 数据类型是一种 12 个字节的结构，以预定义的结构保存日期和时间信息。 可以在块的临时存储器中或者在 DB 中定义 DTL。

尽管以下 BCD 数字格式不能用作数据类型，但它们受转换指令支持。

格式	大小（位）	数字范围	实例
BCD16	16	-999 到 999	123, -123
BCD32	32	-9999999 到 9999999	1234567, -1234567

存储区和寻址

STEP 7 Basic 简化了符号编程。 用户为数据地址创建符号名称或“变量”，作为与存储器地址和 I/O 点相关的 PLC 变量或在代码块中使用的局部变量。 要在用户程序中使用这些变量，只需输入指令参数的变量名称。 为了更好地理解 CPU 的存储区结构及其寻址方式，以下段落将对 PLC 变量所引用的“绝对”寻址进行说明。 CPU 提供了以下几个选项，用于在执行用户程序期间存储数据：

- 全局存储器： CPU 提供了各种专用存储区，其中包括输入 (I)、输出 (Q) 和位存储器 (M)。 所有代码块可以无限制地访问该储存器
- 数据块 (DB)： 可在用户程序中加入 DB 以存储代码块的数据。 从相关代码块开始执行一直到结束，存储的数据始终存在。 “全局”DB 存储所有代码块均可使用的数据，而背景 DB 存储特定 FB 的数据并且由 FB 的参数进行构造。
- 临时存储器： 只要调用代码块，CPU 的操作系统就会分配要在执行块期间使用的临时或本地存储器 (L)。 代码块执行完成后，CPU 将重新分配本地存储器，以用于执行其它代码块。

每个存储单元都有唯一的地址。 用户程序利用这些地址访问存储单元中的信息。

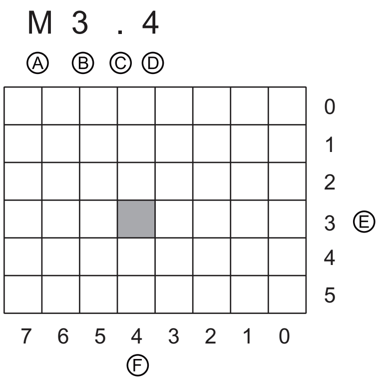
对输入 (I) 或输出 (Q) 存储区（例如 I0.3 或 Q1.7）的引用会访问过程映像。 要立即访问物理输入或输出，请在引用后面添加":P"（例如，I0.3:P、Q1.7:P 或 "Stop:P"）。

仅向输入 (I) 或输出 (Q) 强制写入值。 要强制输入或输出，请在 PLC 变量或地址后面添加":P"。 更多相关信息，请参见 "强制 CPU 中的变量" (页 95)。

存储区	说明	强制	保持性
I 过程映像输入	在扫描周期开始时从物理输入复制	否	否
I_:P[1] （物理输入）	立即读取 CPU、SB 和 SM 上的物理输入点	是	否
Q 过程映像输出	在扫描周期开始时复制到物理输出	否	否
Q_:P[1] （物理输出）	立即写入 CPU、SB 和 SM 上的物理输出点	是	否
M 位存储器	控制和数据存储器	否	是 （可选）
L 临时存储器	存储块的临时数据，这些数据仅在该块的本地范围内有效	否	否
DB 数据块	数据存储器，同时也是 FB 的参数存储器	否	是 （可选）

[1] 要立即访问（或强制）物理输入和物理输出，请在地址或变量后面添加":P"（例如，I0.3:P、Q1.7:P 或 "Stop:P"）。

每个存储单元都有唯一的地址。用户程序利用这些地址访问存储单元中的信息。 下图说明了如何访问一个位（也称为"字节.位"寻址）。 在本例中，存储区和字节地址（M 代表位存储区，3 代表字节 3）通过后面的句点（"."）与位地址（位 4）分隔。



A 存储区标识符

B 字节地址： 字节 3

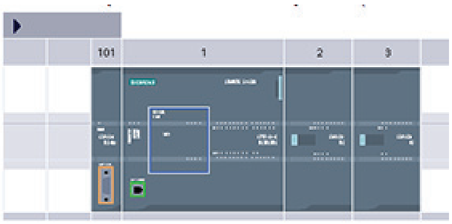
C 分隔符（"字节.位"）

D 位在字节中的位置（位 4，共 8 位）

E 存储区的字节

F 选定字节的位

组态 I/O 的地址




设备概览

模块	插槽	I 地址	Q 地址	类型	订货号
	103				
	102				
RS485_1	101			CM 1241 (RS485)	6ES7 2
▾ PLC_1	1			CPU 1214C DC/DC/DC	6ES7 2
DI14/DO10	1.1	0...1	0...1	DI14/DO10	
AI2	1.2	64...67		AI2	
AO1 x 12 ...	1.3		80...81	AO1 信号板	6ES7 2
HSC_1	1.16	1000...1...		高速计数器 (HSC)	
HSC_2	1.17			高速计数器 (HSC)	
HSC_3	1.18			高速计数器 (HSC)	
HSC_4	1.19			高速计数器 (HSC)	
HSC_5	1.20			高速计数器 (HSC)	
HSC_6	1.21			高速计数器 (HSC)	
Pulse_1	1.32			脉冲发生器 (PTO/P...	
Pulse_2	1.33			脉冲发生器 (PTO/P...	
▸ PROFINET ...	X1			PROFINET 接口	
DI8 x 24VDC_1	2	8		SM 1221 DI8 x 24V...	6ES7 2

向组态画面添加 CPU 和 I/O 模块时，系统会自动分配 I 地址和 Q 地址。

通过在组态画面中选择地址域并键入新编号，可以更改默认寻址设置。数字输入和输出按完整的 8 位字节方式进行分配，无论模块是否使用所有的点。模拟输入和输出按每组 2 点（4 个字节）的方式进行分配。在此实例中，可以将 DI16 的地址改为 2..3 来替代 8..9。工具可以协助您更改大小错误或与其它地址相冲突的地址范围。

图中显示的实例是配有两个 SM 的 CPU 1214C。

4.4 用户程序的执行

CPU 支持以下类型的代码块，使用它们可以创建有效的用户程序结构：

- 组织块 (OB) 是通常包含主程序逻辑的代码块。 OB 对 CPU 中的特定事件作出响应，并可中断用户程序的执行。用于循环执行用户程序的默认组织块 (OB 1) 为用户程序提供基本结构，是唯一一个用户必需的代码块。 其它 OB 执行特定的功能，如处理启动任务、处理中断和错误或以特定的时间间隔执行特定程序代码。
- 功能块 (FB) 是从另一个代码块（OB、FB 或 FC）进行调用时执行的子例程。调用块将参数传递到 FB，并标识可存储特定调用数据或该 FB 实例的特定数据块 (DB)。更改背景 DB 可实现使用一个通用 FB 控制一组设备的运行。 例如，借助包含每个泵或阀门的特定运行参数的不同背景 DB，一个 FB 可控制多个泵或阀。 背景 DB 会保存该 FB 在不同调用或连续调用之间的值，以便能支持异步通信。
- 功能 (FC) 是从另一个代码块（OB、FB 或 FC）进行调用时执行的子例程。FC 不具有相关的背景 DB。调用块将参数传递给 FC。 如果用户程序的其它元素需要使用 FC 的输出值，则必须将这些值写入存储器地址或全局 DB 中。

用户程序、数据及组态的大小受 CPU 中可用装载存储器和工作存储器的限制。对所支持的块数量没有限制；唯一的限制就是存储器大小。

使用 OB 处理事件

CPU 扫描的处理由事件来驱动。 默认事件是启动程序循环 OB 执行的程序循环事件。（用户不需要在程序中使用程序循环 OB。 但是，如果没有程序循环 OB，将不会执行正常的 I/O 更新。 因此就必须通过过程映像来对 I/O 进行读取和写入。）可根据需要启用其它事件。某些事件（例如循环事件）在组态时启用。另一些事件（例如延时事件）在运行时启用。 事件在启用后将连接到相关的 OB。（每个程序循环事件和启动事件都可以连接到多个 OB。）事件发生时，系统将执行该事件的服务例程，即所连接的 OB 以及从该 OB 调用的所有功能。优先级、优先级组以及队列用于确定事件服务例程的处理顺序。

单一来源的未决（排队的）事件数量通过各种事件类型的不同队列加以限制。达到给定事件类型的未决事件限制后，下一个事件将丢失。 每个事件都有一个关联的优先级，而事件优先级分为若干个优先级组，如下表所示。

通常，事件按优先级顺序进行处理（优先级最高的最先进行处理）。优先级相同的事件按“先到先得”的原则进行处理。 OB 开始执行后，如果发生另一个相同或较低优先级组中的事件，则该 OB 的处理不会被中断。 此类事件将排队等待稍后被处理，从而使 CPU 能够完成当前 OB 的执行。

优先级组中的 OB 不会中断属于同一优先级组的其它 OB。 但是，优先级组 2 中的事件将中断优先级组 1 中 OB 的执行，而优先级组 3 中的事件将中断优先级组 1 或 2 中任何 OB 的执行。

事件 (OB)	数量	OB 编号	队列深度	优先级组	优先等级
程序循环	1 个程序循环事件 允许多个 OB	1（默认） 200 或更大	1	1	1
启动	1 个启动事件 [1, 2] 允许多个 OB	100（默认） 200 或更大	1		1
时间延迟	最多 4 个时间事件 [3] 每个事件 1 个 OB	200 或更大	8	2	3
循环	最多 4 个时间事件 [3] 每个事件 1 个 OB	200 或更大	8		4
沿	16 个上升沿事件 16 个下降沿事件 每个事件 1 个 OB	200 或更大	32		5

事件 (OB)	数量	OB 编号	队列深度	优先级组	优先等级
HSC	6 个 CV = PV 事件 6 个方向更改事件 6 个外部复位事件 每个事件 1 个 OB	200 或更大	16		6
诊断错误	1 个事件（仅限 OB 82）	仅限 82	8		9
时间错误	1 个时间错误事件 1 个 MaxCycle 时间事件（仅限 OB 80） 1 个 2xMaxCycle 时间事件	仅限 80	8	3	26 27

[1] 启动事件的特殊情况：启动事件和程序循环事件永远不会同时发生，因为在启动事件运行完成之后才会启动程序循环事件（由操作系统控制）。

[2] 启动事件的特殊情况：只允许诊断错误事件（与 OB 82 关联）中断启动事件。启动事件完成后，其它所有事件排队等待随后被处理。

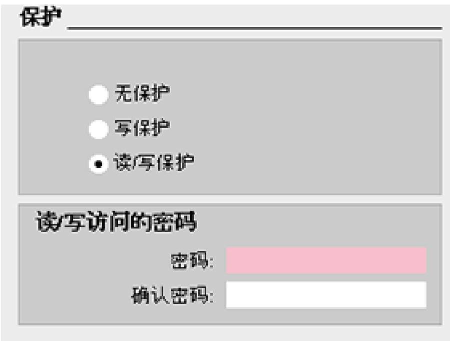
[3] CPU 提供总共 4 个由延时 OB 和循环 OB 共享的时间事件。 用户程序中的延时和循环 OB 数量不能超过 4 个。

较高优先级组中的 OB 会中断较低优先级组中 OB 的执行。 例如，优先级组 2 中的 OB（例如循环中断 OB）会中断程序循环 OB（优先级组 1），而 OB 80（优先级组 3）会中断优先级组 1 或 2 中的任何 OB。但是，同一优先级组中的 OB 不会互相中断。 CPU 会存储处理 OB 期间发生的任何事件。执行完该 OB 后，CPU 随后根据该优先级组内的相对优先等级执行队列中的 OB，并且先处理优先等级较高的事件。 但是，CPU 每次执行完该优先级组中的一个 OB 后，才会开始执行同一个优先级组中的下一个 OB。 处理完中断优先级组的所有事件后，CPU 将返回到较低优先级组中被中断的 OB，并从中断点继续执行该 OB。

假设 CPU 检测到优先级组 3 中的事件（如时间错误事件），则时间错误 OB 将中断优先级组 1（如程序循环 OB）和优先级组 2（如循环 OB）的执行。 CPU 会先执行时间错误 OB，然后再回头执行优先级组 2（如果被中断）或优先级组 1 中被中断的 OB。

4.5 防止对 CPU 或代码块的访问非常容易

CPU 提供了 3 个安全等级，用于限制对特定功能的访问。 为 CPU 组态安全等级和密码时，可以对那些不输入密码就能访问的功能和存储区进行限制。



要组态密码，请按以下步骤操作：

1. 在“设备配置”(Device configuration) 中，选择 CPU。
2. 巡视窗口中，选择“属性”(Properties) 选项卡。
3. 选择“保护”(Protection) 属性以选择保护等级和输入密码。

密码区分大小写。

每个等级都允许在访问某些功能时不使用密码。 CPU 的默认状态是没有任何限制，也没有密码保护。 要限制 CPU 的访问，可以对 CPU 的属性进行组态并输入密码。

通过网络输入密码并不会使 CPU 的密码保护受到威胁。 受密码保护的 CPU 每次只允许一个用户不受限制地进行访问。 密码保护不适用于用户程序指令的执行，包括通信功能。 输入正确的密码便可访问所有功能。

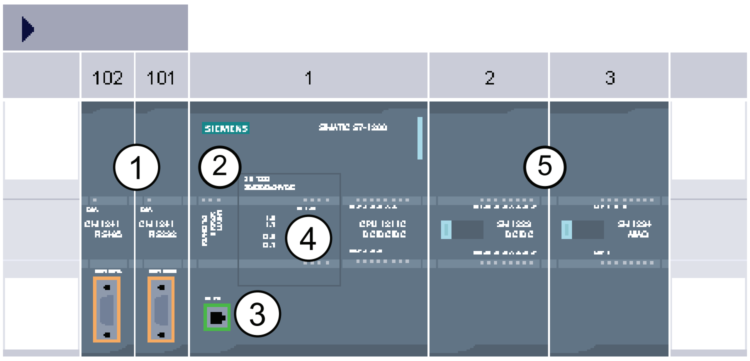
PLC 到 PLC 通信（使用代码块中的通信指令）不受 CPU 中安全等级的限制。 HMI 功能同样也不受限制。

安全等级	访问限制
无保护	允许完全访问，没有密码保护。
写保护	允许 CPU 的只读访问、HMI 访问以及 PLC 到 PLC 通信，没有密码保护。 以下情况下需要密码：修改（写入）CPU 以及更改 CPU 模式 (RUN/STOP)。
读/写保护	允许 HMI 访问和所有形式的 PLC 到 PLC 通信，没有密码保护。 以下情况下需要密码：读取 CPU 中的数据、修改（写入）CPU 以及更改 CPU 模式 (RUN/STOP)。

5 简化了编程概念

5.1 轻松创建设备配置

通过向项目中添加 CPU 和其它模块为 PLC 创建设备配置。



① 通信模块 (CM)： 最多 3 个，分别插在插槽 101、102 和 103 中

② CPU： 插槽 1

③ CPU 的以太网端口

④ 信号板 (SB)： 最多 1 个，插在 CPU 中

⑤ 数字或模拟 I/O 的信号模块 (SM)： 最多 8 个，分别插在插槽 2 到 9 中

CPU 1214C 允许使用 8 个；CPU 1212C 允许使用 2 个；CPU 1211C 不允许使用任何信号模块

要创建设备配置，需向项目中添加设备。

- 在门户视图中，选择“设备和网络”(Devices & Networks) 并单击“添加设备”(Add device)。
- 在项目视图中的项目名称下，双击“添加新设备”(Add new device)。





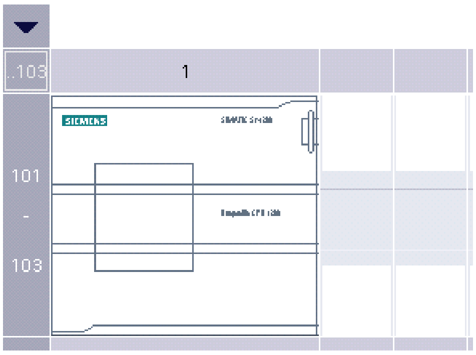
上传现有硬件配置非常简单

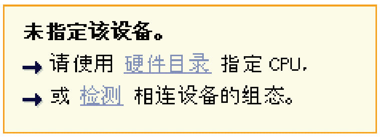


如果已连接到 CPU，则可以将该 CPU（包括所有模块）的组态上传到用户项目中。 只需创建新项目并选择“未指定的 CPU”而不是选择特定的 CPU 即可。 （也可通过从“新手上路”(First steps) 中选择“创建 PLC 程序”(Create a PLC program) 完全跳过设备配置。 STEP 7 Basic 即会自动创建一个未指定的 CPU。）

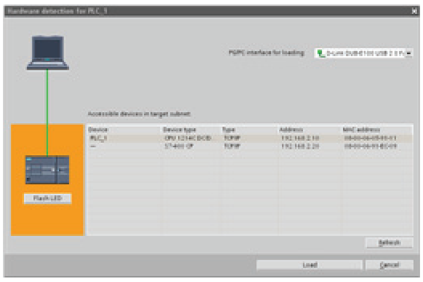
在程序编辑器中，从“在线”(Online) 菜单中选择“硬件检测”(Hardware detection) 命令。

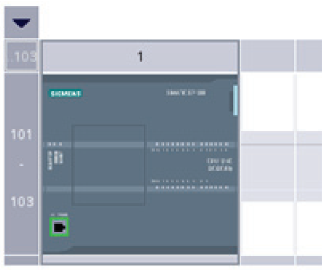
在设备配置编辑器中，选择用于检测所连设备组态的选项。





从在线对话框中选择 CPU 之后，STEP 7 Basic 会上传 CPU 以及所有模块（SM、SB 或 CM）的硬件配置。 随后可以为 CPU 和模块 (页 44)组态参数。

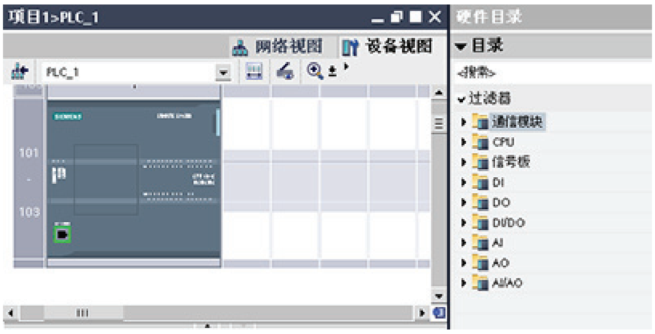




将 CPU 添加到组态中



通过将 CPU 插入到项目中创建设备配置。 在“添加新设备”(Add a new device) 对话框中选择 CPU 并单击“确定”(OK)，将 CPU 添加到项目中。



设备视图显示了 CPU 和机架。



通过在设备视图中选择 CPU，可在巡视窗口中显示 CPU 属性。 使用这些属性 组态 CPU 的运行参数 (页 44)。

说明

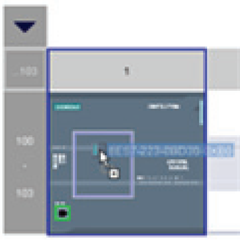
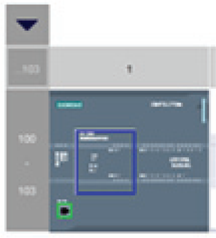
CPU 不具有预组态的 IP 地址。设备配置期间必须为 CPU 手动分配 IP 地址。如果 CPU 连接到网络上的路由器，则也应输入路由器的 IP 地址。

将设备添加到组态中

使用硬件目录将模块添加到 CPU。有三种类型的模块：

- 信号板 (SB) 仅为 CPU 提供几个附加的 I/O 点。SB 安装在 CPU 的前端。
- 信号模块 (SM) 提供附加的数字或模拟 I/O 点。这些模块连接在 CPU 右侧。
- 通信模块 (CM) 为 CPU 提供附加的通信端口（RS232 或 RS485）。这些模块连接在 CPU 左侧。

要将模块插入到硬件配置中，可在硬件目录中选择模块，然后双击该模块或将其拖到高亮显示的插槽中。

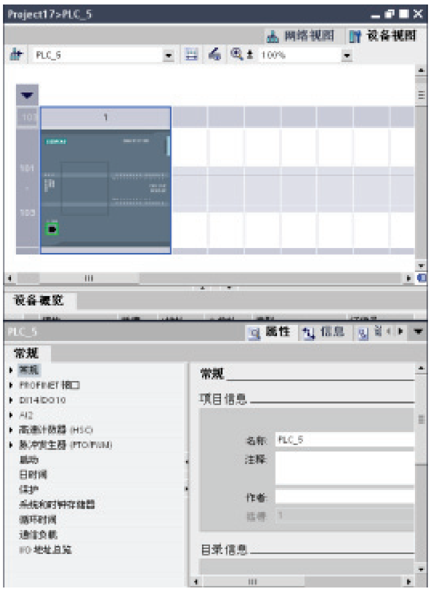
模块	选择模块	插入模块	结果
SM			
SB			
CM			

组态网络连接

使用设备配置的“网络视图”(Network view) 在项目中的各个设备之间创建网络连接。 创建网络连接之后，使用巡视窗口的“属性”(Properties) 选项卡组态网络的参数。

“设备配置”的网络视图	说明
	选择“网络视图”(Network view) 以显示要连接的设备。
	选择一个设备上的端口，然后将连接拖到第二个设备的端口上。
	释放鼠标按钮以创建网络连接。

5.1.1 组态 CPU 和模块的运行



要组态 CPU 的运行参数，请在设备视图中选择该 CPU，并使用巡视窗口的“属性”(Properties) 选项卡。

- CPU 的 PROFINET IP 地址和时间同步
- CPU 在关机到开机切换之后的启动特性
- 本地（板载）数字和模拟 I/O、高速计数器 (HSC) 和脉冲发生器
- 系统时钟（时间、时区和夏令时）
- 访问 CPU 所涉及的读/写保护和密码
- 最大循环时间或固定最小循环时间以及通信负载

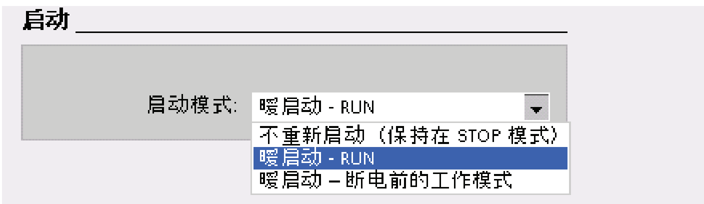
组态 CPU 从 STOP 到 RUN 的运行

只要工作状态从 STOP 切换到 RUN，CPU 就会清除过程映像输入、初始化过程映像输出并处理启动 OB。（因此，启动 OB 中的指令此时对过程映像输入进行任何读访问，读取到的只有零，而不是当前物理输入值。）要在启动模式下读取物理输入的当前状态，必须先执行立即读取操作。 接着再执行启动 OB 以及任何相关的 FC 和 FB。 如果存在多个启动 OB，则按照 OB 编号依次执行各启动 OB，OB 编号最小的先执行。

在启动过程中，CPU 还会执行以下任务。

- 在启动阶段，对中断进行排队但不加以处理
- 在启动阶段，不执行任何循环时间监视
- 在启动模式下，可以更改 HSC（High-Speed Counter，高速计数器）、PWM（Pulse-Width Modulation，脉冲宽度调制）以及 PtP（Point-to-Point communication，点对点通信）模块的组态
- 只有在 RUN 模式下才会真正运行 HSC、PWM 和点对点通信模块

执行完启动 OB 后，CPU 将进入 RUN 模式并在连续的扫描周期内处理控制任务。



使用 CPU 属性组态 CPU 在上电循环后的启动方式： 处于 STOP 模式、RUN 模式还是上一个模式（上电循环之前）。

CPU 在进入 RUN 模式前执行暖启动。 暖启动会将所有非保持性存储器复位为默认初始值，但 CPU 会保留保持性存储器中存储的当前值。

说明

下载完成后 CPU 总是会执行冷启动

每次下载完项目元素（例如程序块、数据块或硬件配置），CPU 都会在下一次转换为 RUN 模式时先执行冷启动。 除清除输入、初始化输出以及清除非保持性存储器之外，冷启动还会清除保持性存储区。

在紧随下载的冷启动完成之后，所有随后的 STOP 到 RUN 转换均会执行暖启动（不会清除保持性存储器）。

组态 I/O 和通信模块的运行

要组态信号模块 (SM)、信号板 (SB) 或通信模块 (CM) 的运行参数，请在设备视图中选择相应模块，并使用巡视窗口的“属性”(Properties) 选项卡。



信号模块 (SM) 和信号板 (SB)

- 数字量 I/O： 组态各个输入，如用于沿检测和“脉冲捕捉”（瞬时脉冲之后停留） 组态输出以在 RUN 模式转换到 STOP 模式时使用冻结值或替换值。
- 模拟量 I/O： 组态各个输入的参数（如电压/电流、范围和平滑化），并启用下溢或上溢诊断。 组态各个模拟量输出的参数并启用诊断，例如，短路（对于电压输出）或溢出值诊断。
- 诊断地址： 组态用于设置模块的输入和输出的起始地址。



通信模块 (CM)

- 端口组态： 组态通信参数，如波特率、奇偶校验、数据位、停止位以及等待时间
- 发送和接收消息： 组态与发送和接收数据相关的选项（例如，消息起始参数和消息结束参数）

也可以通过用户程序更改这些组态参数。

5.1.2 组态 CPU 的 IP 地址

因为 CPU 没有预组态的 IP 地址，所以必须手动分配 IP 地址。 在组态 CPU 的属性时组态 PROFINET 接口的 IP 地址与其它参数。

- 在 PROFINET 网络中，制造商会为每个设备都分配一个唯一的“介质访问控制”地址（MAC 地址）以进行标识。 每个设备也都必须具有一个 IP 地址。
- 子网是已连接的网络设备的逻辑分组。 掩码（又称为子网掩码或网络掩码）定义子网的边界。 不同子网间的唯一连接通过路由器实现。 路由器是 LAN 之间的链接，它依靠 IP 地址传送和接收数据包。

将 IP 地址下载到 CPU 之前，必须先确保计算机的 IP 地址与编程设备的 IP 地址相匹配。

可以使用 STEP 7 Basic 确定编程设备的 IP 地址：

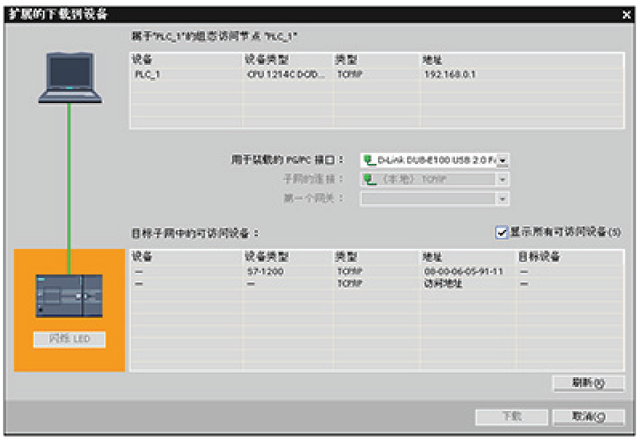
1. 在项目树中右键单击“在线访问”(Online access) 文件夹以显示快捷菜单。
2. 选择“属性”(Properties) 命令。



该对话框显示编程设备的设置。

CPU 的 IP 地址必须与编程设备的 IP 地址和子网掩码兼容。 请咨询网络专家以确定 CPU 的 IP 地址和子网掩码。

确定 CPU 的 IP 地址和子网掩码之后，输入 CPU 和路由器（如果适用）的 IP 地址。 请参见*《S7-1200 系统手册》*以获取更多信息。



完成组态后，将项目下载到 CPU 中。

在下载项目前组态 CPU 和路由器（如果适用）的 IP 地址。

5.2 轻松设计用户程序

创建处理自动化任务的用户程序时，需要将程序指令插入代码块（OB、FB 或 FC）中。

OB 是用于针对您的应用构建和组织用户程序的代码块。 对于许多应用，都采用一个不断循环的 OB（如程序循环 OB 1）来包含程序逻辑。 除程序循环 OB 外，CPU 还提供其它执行特定功能的 OB，例如执行启动任务、处理中断和错误或者以特定的时间间隔执行特定程序代码。 每个 OB 都对 CPU 中的一个特定事件作出响应，并能根据预定义的优先级组和等级中断用户程序的执行。

FB 是从另一个代码块（OB、FB 或 FC）进行调用时执行的子例程。调用块将参数传递到 FB，并标识可存储特定调用数据或该 FB 实例的特定数据块 (DB)。更改背景 DB 可实现使用一个通用 FB 控制一组设备的运行。 例如，借助包含每个泵或阀门的特定运行参数的不同背景 DB，一个 FB 可控制多个泵或阀。 背景 DB 会保存该 FB 在不同调用或连续调用之间的值，以便能支持异步通信。

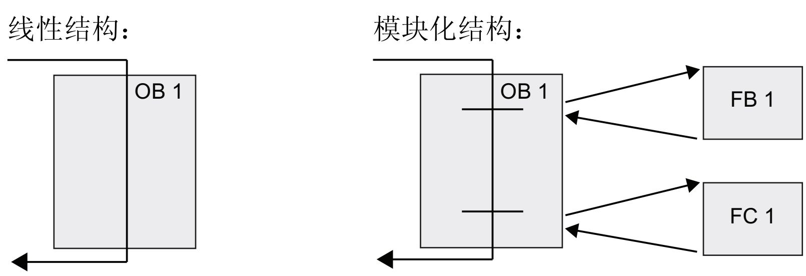
FC 是从另一个代码块（OB、FB 或 FC）进行调用时执行的子例程。FC 不具有相关的背景 DB。调用块将参数传递给 FC。FC 中的输出值必须写入存储器地址或全局 DB 中。

为用户程序选择结构类型

根据实际应用要求，可选择线性结构或模块化结构来创建用户程序。

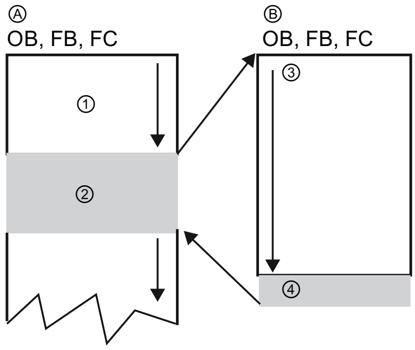
线性程序按顺序逐条执行处理自动化任务的所有指令。 通常，线性程序将所有程序指令都放入一个程序循环 OB (OB 1) 中以循环执行该程序。

模块化程序调用可执行特定任务的特定代码块。 要创建模块化结构，需要将复杂的自动化任务划分为与过程所执行的功能任务相对应的更小的次级任务。每个代码块都为各个次级任务提供程序段。通过从另一个块中调用其中一个代码块来构建程序。



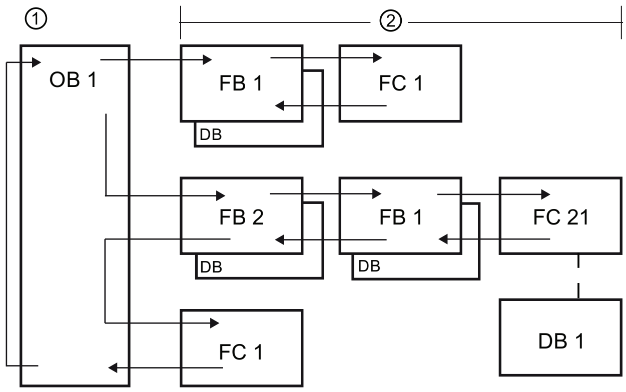
通过设计 FB 和 FC 执行通用任务，可创建模块化代码块。 然后可通过由其它代码块调用这些可重复使用的模块来构建用户程序。调用块将设备特定的参数传递给被调用块。当一个代码块调用另一个代码块时，CPU 会执行被调用块中的程序代码。执行完被调用块后，CPU 会继续执行调用块，并继续执行该块调用之后的指令。

A	调用块
B	被调用（或中断）块
①	程序执行
②	用于触发其它块执行的指令或事件
③	程序执行
④	块结束（返回到调用块）



可嵌套块调用以实现更加模块化的结构。

①	循环开始
②	嵌套深度 在本例中，嵌套深度为 4： 程序循环 OB 加 3 层对代码块的调用。



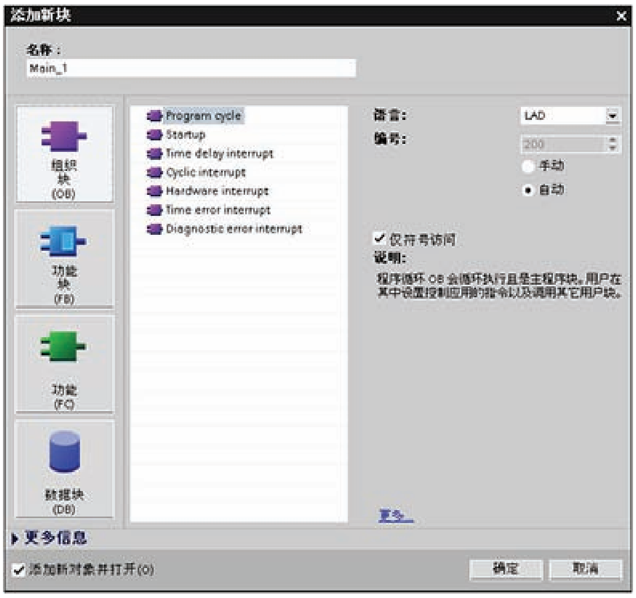
通过创建可在用户程序中重复使用的通用代码块，可简化用户程序的设计和实现。

- 可为标准任务创建能够重复使用的代码块，如用于控制泵或电机。也可以将这些通用代码块存储在可由不同的应用或解决方案使用的库中。
- 将用户程序构建到与功能任务相关的模块化组件中，可使程序的设计更易于理解和管理。 模块化组件不仅有助于标准化程序设计，也有助于使更新或修改程序代码更加快速和容易。
- 创建模块化组件可简化程序的调试。通过将整个程序构建为一组模块化程序段，可在开发每个代码块时测试其功能。
- 利用与特定功能任务相关的模块化设计，可以减少对已完成的应用程序进行调试所需的时间。

5.2.1 使用 OB 组织用户程序

组织块为程序提供结构。 它们充当操作系统和用户程序之间的接口。 OB 是由事件驱动的。 事件（如诊断中断或时间间隔）会使 CPU 执行 OB。某些 OB 预定义了起始事件和行为。

程序循环 OB 包含用户主程序。 用户程序中可包含多个程序循环 OB。 RUN 模式期间，程序循环 OB 以最低优先级等级执行，可被其它各种类型的程序处理中断。 （启动 OB 不会中断程序循环 OB，因为 CPU 在进入 RUN 模式之前执行启动 OB。） 完成处理程序循环 OB 后，CPU 会立即重新执行程序循环 OB。 该循环处理是用于 PLC 的“正常”处理类型。 对于许多应用，整个用户程序位于一个 OB 中，如默认的程序循环 OB 1。



可创建其它 OB 以执行特定的功能，如执行启动任务、处理中断和错误或以特定的时间间隔执行特定程序代码。

使用“添加新块”(Add new block) 对话框在用户程序中创建新的 OB。

CPU 根据 分配给每个 OB 的优先级 (页 35)来确定处理中断事件的顺序。

在某等级的 OB 内创建附加 OB： 可为用户程序创建多个 OB，甚至可以为程序循环和启动 OB 等级创建。 使用“添加新块”(Add new block) 对话框创建 OB。输入 OB 的名称并提供 200 或更大的数作为 OB 编号。

如果为用户程序创建了多个程序循环 OB，则 CPU 会按数字顺序从主程序循环 OB（默认为 OB 1）开始执行每个程序循环 OB。 例如，当第一个程序循环 OB (OB 1) 完成后，CPU 将执行第二个程序循环 OB（例如 OB 200）。

组态 OB 的运行



可修改 OB 的运行参数。例如，可为延时 OB 或循环中断 OB 组态时间参数。

5.2.2 FB 和 FC 使模块化任务编程变得很轻松

功能 (FC) 与子例程类似。 FC 是通常对一组输入值执行特定运算的代码块。 FC 将此运算结果存储在存储单元中。 使用 FC 可执行以下任务：

- 执行标准和可重复使用的运算，例如数学计算。
- 执行功能任务，如通过使用位逻辑运算进行单独控制。

也可以在程序中的不同位置多次调用 FC。 此重复使用简化了对经常重复发生的任务的编程。

与 FB 不同，FC 不具有相关的背景 DB。 FC 使用其临时存储器 (L) 保存用于计算运算的数据。 不保存临时数据。 要存储数据以备 FC 执行完成后使用，可将输出值赋给全局存储单元，如 M 存储器或全局 DB。

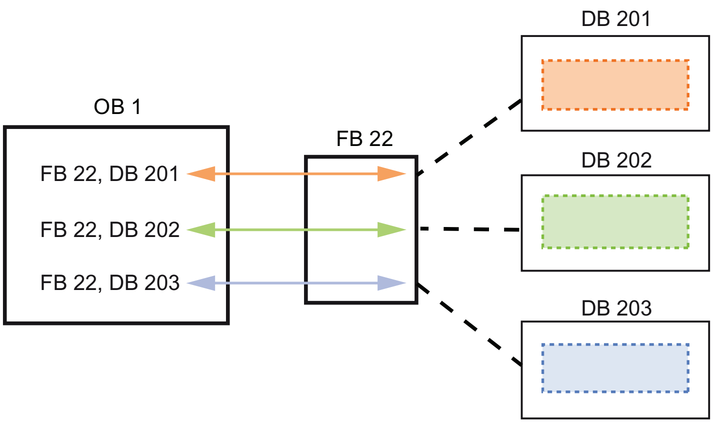
功能块 (FB) 与带存储器的子例程类似。 FB 是可通过块参数以编程方式实现其调用的代码块。 FB 将输入 (IN)、输出 (OUT) 和输入/输出 (IN_OUT) 参数存储在数据块 (DB) 或“背景”DB 中的变量存储器内。 背景 DB 提供与 FB 的实例（或调用）关联的一块存储区并在 FB 完成后存储数据。

用户通常使用 FB 控制在一个扫描周期内未完成其运行的任务或设备的运行。 要存储运行参数以便从一个扫描快速访问到下一个扫描，用户程序中的每一个 FB 都应具有一个或多个背景 DB。 调用 FB 时，也会打开存储块参数的值以及用于该 FB 调用或“背景”的静态局部数据的背景 DB。 FB 完成后，背景 DB 将存储这些值。

可以给 FB 接口中的参数赋初值。 这些值将传送到相关的背景 DB 中。 如果未分配参数，将使用当前存储在背景 DB 中的值。 某些情况下，必须分配参数。

可将不同的背景 DB 与 FB 的不同调用进行关联。 通过背景 DB 可使用一个通用 FB 控制多个设备。 通过使一个代码块对 FB 和背景 DB 进行调用，来构建程序。 然后，CPU 执行该 FB 中的程序代码，并将块参数和静态局部数据存储在背景 DB 中。 FB 执行完成后，CPU 会返回到调用该 FB 的代码块中。 背景 DB 保留该 FB 实例的值。 通过设计用于通用控制任务的 FB，可对多个设备重复使用 FB，方法是：为 FB 的不同调用选择不同的背景 DB。

下图显示了三次调用同一个 FB 的 OB，方法是针对每次调用使用一个不同的数据块。 该结构使一个通用 FB 可以控制多个相似的设备（如电机），方法是在每次调用时为各设备分配不同的背景数据块。 每个背景 DB 存储单个设备的数据（如速度、加速时间和总运行时间）。 在此实例中，FB 22 控制三个独立的设备，其中 DB 201 用于存储第一个设备的运行数据，DB 202 用于存储第二个设备的运行数据，DB 203 用于存储第三个设备的运行数据。



5.2.3 数据块为程序数据提供了便捷的存储方式

在用户程序中创建数据块 (DB) 以存储代码块的数据。用户程序中的所有程序块都可访问全局 DB 中的数据，而背景 DB 仅存储特定功能块 (FB) 的数据。

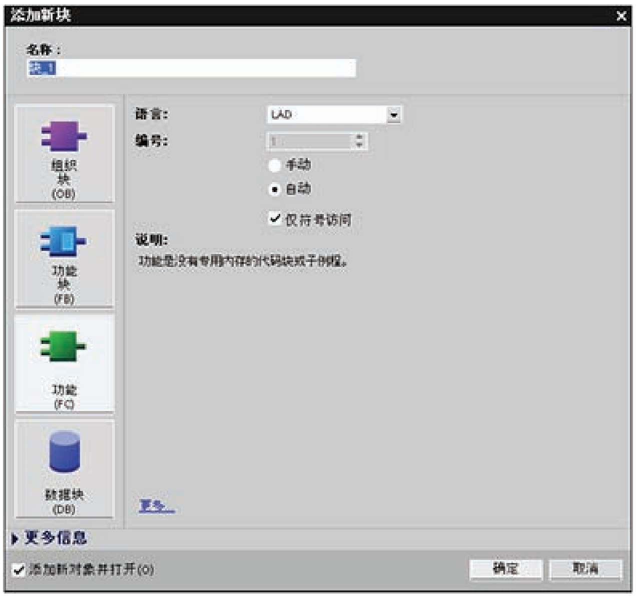
用户程序可将数据存储在 CPU 的专用存储区中，如输入 (I)、输出 (Q) 和位存储器 (M)。此外，可使用数据块 (DB) 快速访问存储在程序本身中的数据。可将 DB 定义为只读。

当数据块关闭或相关代码块的执行结束时，DB 中存储的数据不会被删除。有两种类型的 DB：

- 全局 DB 存储程序中代码块的数据。任何 OB、FB 或 FC 都可访问全局 DB 中的数据。
- 背景 DB 存储特定 FB 的数据。背景 DB 中数据的结构反映了 FB 的参数（Input、Output 和 InOut）和静态数据。 FB 的临时存储器不存储在背景 DB 中。

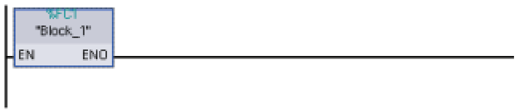
尽管背景 DB 反映特定 FB 的数据，然而任何代码块都可访问背景 DB 中的数据。

创建新代码块



1. 打开“程序块”(Program blocks) 文件夹。
2. 双击“添加新块”(Add new block)。
3. 在“添加新块”(Add new block) 对话框中，单击“功能 (FC)”(Function (FC)) 图标。
4. 通过从下拉菜单中选择“LAD”为 FC 指定编程语言。
5. 单击“确定”(OK) 将块添加到项目中。

选中“添加新对象并打开”(Add new and open) 选项（默认），在项目视图中打开代码块。



用户可以很容易地使用用户程序中的任何代码块（OB、FB 或 FC）调用 CPU 中的 FB 或 FC。

1. 打开将调用另一个块的代码块。
2. 在项目树中，选择要调用的代码块。
3. 将该块拖到所选程序段中以创建一个 Call 指令。

说明

用户程序无法调用 OB，因为 OB 是 由事件驱动的 (页 35)。 CPU 接收到事件时，将开始执行 OB 进行响应。

5.3 轻松使用功能强大的编程语言

可以在 LAD（梯形图）或 FBD（功能块图）编程语言之间做出选择。

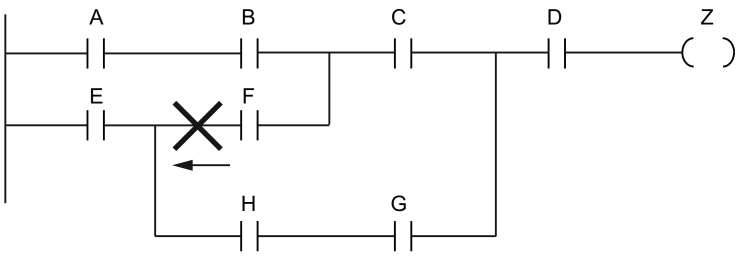
LAD 是一种图形编程语言。它使用基于电路图的表示法。要创建复杂运算逻辑，可插入分支以创建并行电路的逻辑。并行分支向下打开或直接连接到电源线。用户可向上终止分支。 LAD 还为多种功能（如数学运算、定时器、计数器和移动）提供“功能框”指令。



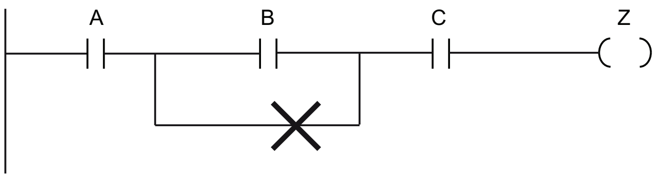
电路图的元件（如常闭触点、常开触点和线圈）相互连接构成程序段。

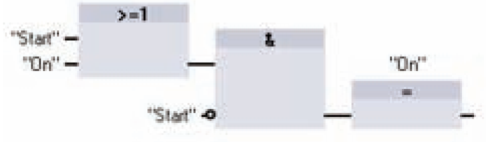
创建 LAD 程序段时请注意以下规则：

- 每个 LAD 程序段都必须使用线圈或功能框指令来终止。不要使用比较指令或沿检测（上升沿或下降沿）指令终止程序段。
- 不能创建可能导致反向能流的分支。



- 不能创建可能导致短路的分支。





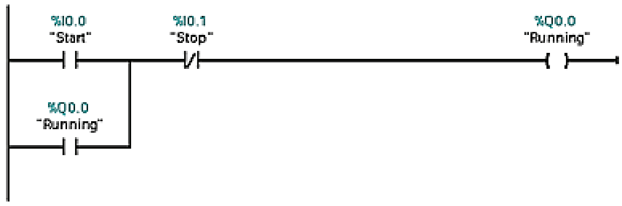
与 LAD 一样，FBD 也是一种图形编程语言。逻辑表示法以布尔代数中使用的图形逻辑符号为基础。

算术函数和其它复杂函数可直接结合逻辑框表示。要创建复杂运算的逻辑，请在功能框之间插入并行分支。

5.3.1 提供您所期望的基本指令

位逻辑指令

位逻辑指令的基础是触点和线圈。 触点读取位的状态，而线圈则将操作的状态写入到位中。

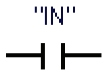


触点可测试位的二进制状态，结果是在接通 (1) 时“有能流”，在断开 (0) 时“没有能流”。

线圈的状态反映前导逻辑的状态。

如果在多个程序段中使用地址相同的线圈，则用户程序中最后一次运算的结果将决定该地址的值状态。

常开触点



常闭触点



在赋的位值为 1 时，常开触点将闭合 (ON)。

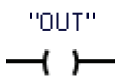
在赋的位值为 0 时，常闭触点将闭合 (ON)。

位逻辑运算的基本结构为 AND 逻辑或 OR 逻辑。 以串联方式连接的触点创建 AND 逻辑程序段。 以并联方式连接的触点创建 OR 逻辑程序段。

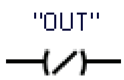
可将触点相互连接，创建用户自己的组合逻辑。 如果用户指定的输入位使用存储器标识符 I（输入）或 Q（输出），则从过程映像寄存器中读取位值。 控制过程中的物理触点信号会连接到 PLC 上的输入端子。 CPU 扫描已连接的输入信号并更新过程映像输入寄存器中的相应状态值。

通过在输入变量后加上“:P”（例如，“Motor_Start:P”或“I3.4:P”），可指定立即读取物理输入。 对于立即读取，将直接从物理输入读取位数据值，而不是从过程映像中读取。 立即读取不会更新过程映像。

输出线圈



反向输出线圈



- 如果有能流通过输出线圈，则输出位设置为 1。
- 如果没有能流通过输出线圈，则输出线圈位设置为 0。
- 如果有能流通过反向输出线圈，则输出位设置为 0。
- 如果没有能流通过反向输出线圈，则输出位设置为 1。

线圈输出指令写入输出位的值。 如果用户指定的输出位使用存储器标识符 Q，则 CPU 接通或断开过程映像寄存器中的输出位，同时设置与能流状态相应的指定位。 控制执行器的输出信号连接到 PLC0 的输出端子。 在 RUN 模式下，CPU 系统将扫描输入信号，并根据程序逻辑处理输入状态，然后通过在过程映像输出寄存器中设置新的输出状态值进行响应。 在每个程序执行循环之后，CPU 都会将存储在过程映像寄存器中的新输出状态响应传送到已连接的输出端子。

通过在输出变量后加上“:P”（例如“Motor_On:P”或“Q3.4:P”），可指定立即写入物理输出。 对于立即写入，会将位数据值写入到过程映像输出并直接写入到物理输出。

线圈并不局限于在程序段结尾使用。 可以在 LAD 程序段的梯级中间以及触点或其它指令之间插入线圈。

NOT 触点反相器 (LAD)



带一个反向逻辑输入的 AND 功能框 (FBD)



带反向逻辑输入和输出的 AND 功能框 (FBD)

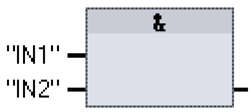


LAD NOT 触点用于对能流输入的逻辑状态取反。

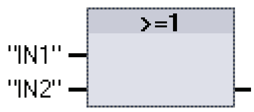
- 如果没有能流流入 NOT 触点，则会有能流流出。
- 如果有能流流入 NOT 触点，则没有能流流出。

对于 FBD 编程，可从“收藏夹”(Favorites) 工具栏或指令树中拖动“二进制输入取反”(Negate binary input) 工具，然后将其放置在输入或输出端以在该功能框连接器上创建逻辑反相器。

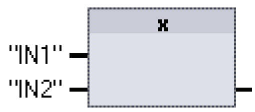
AND 功能框 (FBD)



OR 功能框 (FBD)



XOR 功能框 (FBD)



- AND 功能框的所有输入必须都为 TRUE，输出才为 TRUE。
- OR 功能框只要有一个输入为 TRUE，输出就为 TRUE。
- XOR 功能框必须有奇数个输入为 TRUE，输出才为 TRUE。

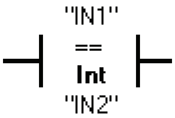
在 FBD 编程中，LAD 的触点程序段由与 (&)、或 (>=1) 和异或 (x) 功能框程序段表示，可在其中为功能框输入和输出指定位值。 也可以连接到其它逻辑框，创建用户自己的逻辑组合。 在程序段中放置功能框后，可从“收藏夹”(Favorites) 工具栏或指令树中拖动“插入二进制输入”(Insert binary input) 工具，然后将其放置在功能框的输入侧以添加更多输入。 也可以右键单击功能框输入连接器并选择“插入输入”(Insert input)。

功能框输入和输出可连接到其它逻辑框，也可输入未连接输入的位地址或位符号名称。 执行功能框指令时，当前输入状态会应用到二进制功能框逻辑，如果为真，功能框输出将为真。

比较指令

使用比较指令可比较两个数据类型相同的值。 比较结果为 TRUE 时，触点将被激活 (LAD) 或者功能框输出为 TRUE (FBD)。

LAD



FBD

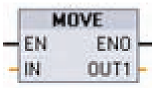


在程序编辑器中单击该指令后，可以从下拉菜单中选择比较类型和数据类型。

关系类型	满足以下条件时比较结果为真：
==	IN1 等于 IN2
<>	IN1 不等于 IN2
>=	IN1 大于或等于 IN2
<=	IN1 小于或等于 IN2
>	IN1 大于 IN2
<	IN1 小于 IN2

移动和块移动指令

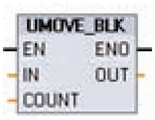
使用移动指令将数据元素复制到新的存储器地址并从一种数据类型转换为另一种数据类型。 移动过程不会更改源数据。



MOVE 可将存储在指定地址的数据元素复制到新地址。



MOVE_BLK（可中断移动）可将数据元素块复制到新地址

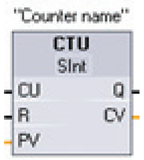
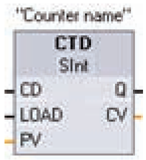
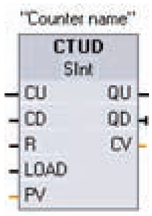


UMOVE_BLK（不可中断移动）可将数据元素块复制到新地址

- MOVE 指令将单个数据元素从 IN 参数指定的源地址复制到 OUT 参数指定的目标地址。
- MOVE_BLK 和 UMOVE_BLK 指令具有附加的 COUNT 参数。 COUNT 指定要复制的数据元素个数。 每个被复制元素的字节数取决于 PLC 变量表中分配给 IN 和 OUT 参数变量名称的数据类型。

计数器

可使用计数器指令对内部程序事件和外部过程事件进行计数。 每个计数器都使用数据块中存储的结构来保存计数器数据。 用户在编辑器中放置计数器指令时分配相应的数据块。 这些指令使用软件计数器，软件计数器的最大计数速率受其所在的 OB 的执行速率限制。

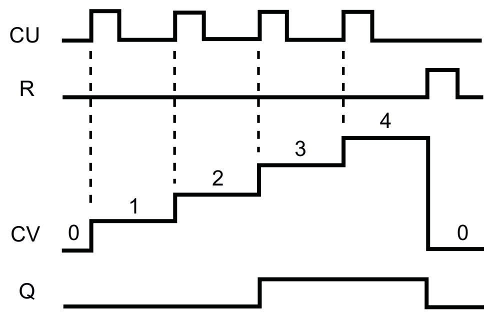
CTU 用于加计数。	CTD 用于减计数。	CTUD 用于加计数和减计数。	
			从计数器名称下的下拉列表中选择计数值数据类型。

用户程序中可以使用的计数器数仅受 CPU 存储器容量限制。 计数器占用以下存储器空间：

- 对于 SInt 或 USInt 数据类型，计数器指令占用 3 个字节。
- 对于 Int 或 UInt 数据类型，计数器指令占用 6 个字节。
- 对于 DInt 或 UDInt 数据类型，计数器指令占用 3 个字节。

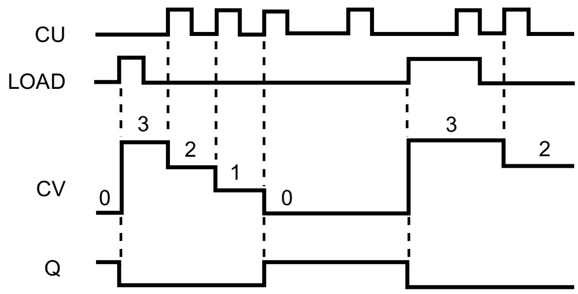
当参数 CU 的值从 0 变为 1 时，CTU 会使计数值加 1。下图显示了计数值为无符号整数时的 CTU 时序图（其中，PV = 3）。

- 如果参数 CV（当前计数值）的值大于或等于参数 PV（预设计数值）的值，则计数器输出参数 Q = 1。
- 如果复位参数 R 的值从 0 变为 1，则当前计数值重置为 0。



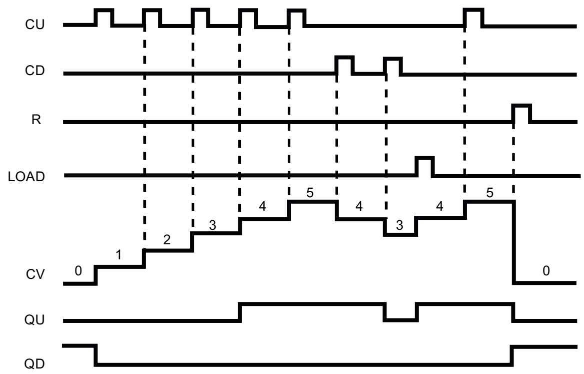
当参数 CD 的值从 0 变为 1 时，CTD 会使计数值减 1。下图显示了计数值为无符号整数时的 CTD 时序图（其中，PV = 3）。

- 如果参数 CV（当前计数值）的值等于或小于 0，则计数器输出参数 Q = 1。
- 如果参数 LOAD 的值从 0 变为 1，参数 PV（预设值）的值将作为新的 CV（当前计数值）装载到计数器。



当加计数 (CU, Count Up) 或减计数 (CD, Count Down) 输入的值从 0 变为 1 时，CTUD 会使计数值加 1 或减 1。 下图显示了计数值为无符号整数时的 CTUD 时序图（其中，PV = 4）。

- 如果参数 CV（当前计数值）的值等于或大于参数 PV（预设值）的值，则计数器输出参数 QU = 1。
- 如果参数 CV 的值小于或等于 0，则计数器输出参数 QD = 1。
- 如果参数 LOAD 的值从 0 变为 1，则参数 PV（预设值）的值将作为新的 CV（当前计数值）装载到计数器。 如果复位参数 R 的值从 0 变为 1，则当前计数值重置为 0。

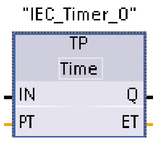


定时器

使用定时器指令可创建编程的时间延迟：

- TP： 脉冲定时器可生成具有预设宽度时间的脉冲。
- TON： 接通延迟定时器输出 Q 在预设的延时过后设置为 ON。
- TOF： 关断延迟定时器输出 Q 在预设的延时过后重置为 OFF。
- TONR： 保持型接通延迟定时器输出在预设的延时过后设置为 ON。 在使用输入 R 重置所消耗的时间之前，会一直累加多个定时时段内耗用的时间。
- RT： 通过清除存储在指定定时器背景数据块中的时间数据来重置定时器。

TP、TON 和 TOF 定时器具有相同的输入和输出参数。



TONR 定时器具有附加的复位输入参数 R。



RT 指令可重置指定定时器的定时器数据。

“定时器名称”

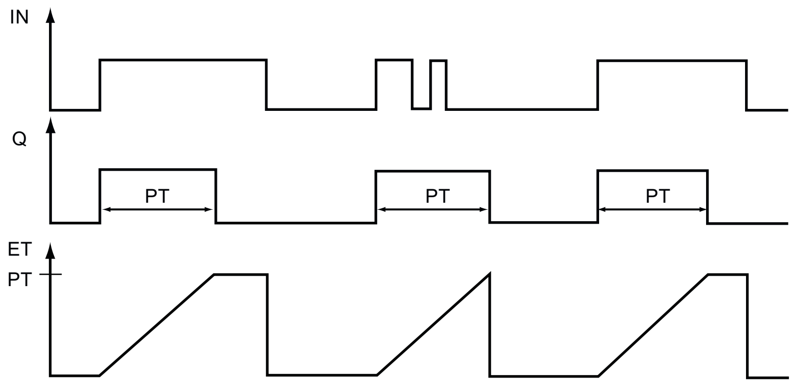
----[RT]----

用户程序中可以使用的定时器数仅受 CPU 存储器容量限制。 每个定时器占用 16 个字节的存储器空间：

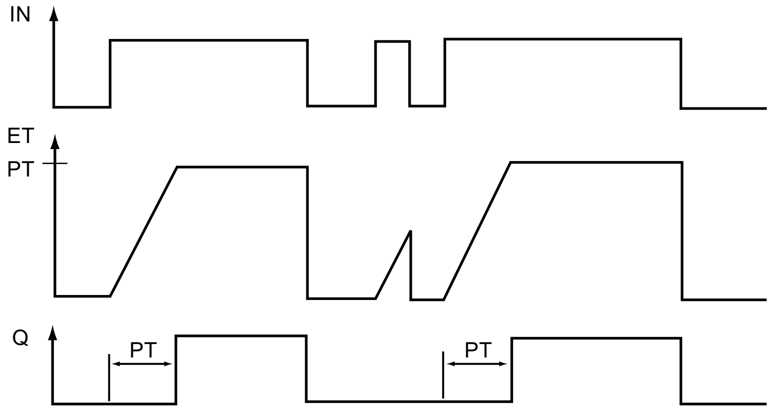
每个定时器都使用一个存储在数据块中的结构来保存定时器数据。 在编辑器中放置定时器指令时即可分配该数据块。 在功能块中放置定时器指令后，可以选择多重背景数据块选项，各数据结构的定时器结构名称可以不同，但定时器数据包含在单个数据块中，而且每个定时器不必使用一个单独的数据块。 这样可减少处理定时器所需的处理时间和数据存储空间。 在共享的多重背景数据块中的定时器数据结构之间不存在交互作用。

TP 定时器

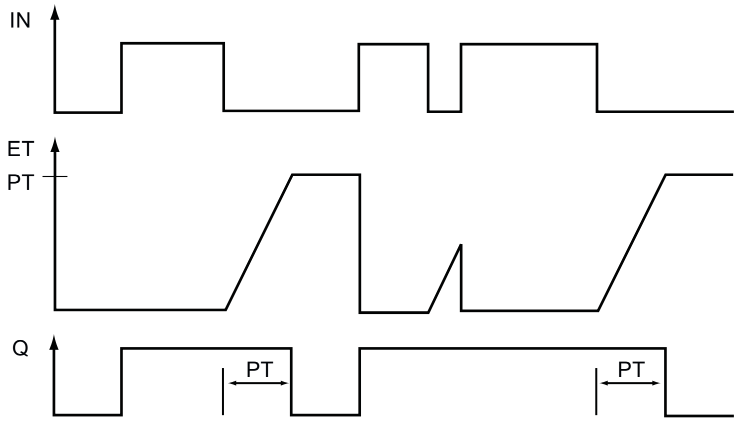
脉冲时序图



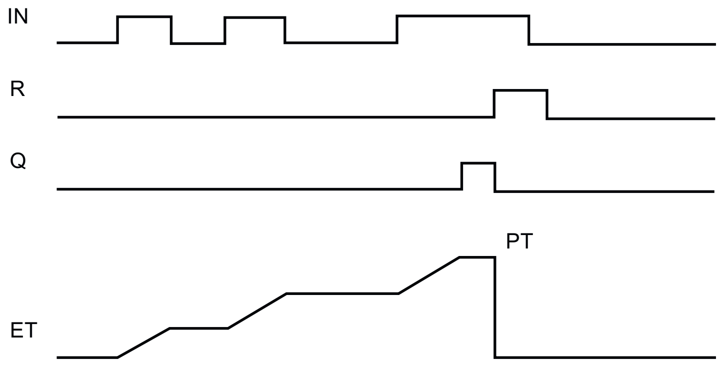
TON 定时器
接通延迟时序图



TOF 定时器
关断延迟时序图



TONR 定时器
保持型接通延迟时序图



S7-1200 提供了功能强大的指令

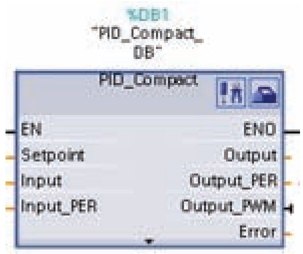
除基本指令外，S7-1200 还提供了其它一些功能强大的指令集，可帮助您轻松解决复杂控制应用方面的问题。 以下指令只是一些集成在 S7-1200 中的强大功能的实例。

CTRL_PWM 指令： CTRL_PWM 脉冲宽度调制 (PWM, Pulse Width Modulation) 指令可提供占空比可变的固定循环时间输出。 PWM 输出以指定频率（循环时间）启动之后将连续运行。 脉冲宽度会根据需要进行变化。



更多相关信息，请参见 脉冲宽度调制 (页 83)的说明。

PID_Compact 指令： PID（比例/积分/微分）控制使用 PID 算法计算反馈和设定点值的差，并将结果输出到执行器（例如，加热器或变频器），以保持设定点。 PID_ Compact 指令可用来提供可在自动和手动模式下自我优化调节的 PID 控制器。



以恒定的采样时间间隔（最好在循环中断 OB 中）执行 PID_Compact 指令。

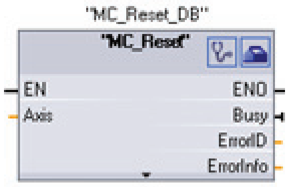
PID_Compact 指令用于测量两次调用之间的时间间隔并评估监视采样时间的结果。 每次进行模式切换时以及初始启动期间都会生成采样时间的平均值。 该值用作监视功能的参考并用于块中的计算。 监视包括两次调用之间的当前测量时间和定义的控制器采样时间的平均值。

模式	说明
未激活	首次下载用户程序后，PID 控制器仍处于“未激活”工作模式。 在这种情况下，将执行调试窗口中的“初始启动期间的自调节”。 在运行期间，如果发生错误或在调试窗口中单击“控制器停止”(Controller stop) 图标，PID 控制器会切换为“未激活”工作模式。
自调节	在调试窗口中调用此功能时，将执行“初始启动期间的自调节”或“工作点中的自调节”工作模式。
自动模式	在自动模式下，PID_Compact 指令会按照指定的参数更正控制回路。
手动模式	如果 PID 控制器在手动模式下运行，则可手动设置受控变量。

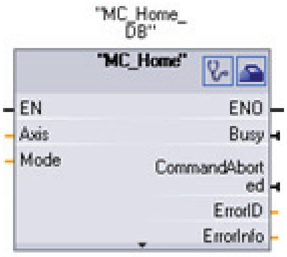
运动控制指令： 运动控制指令使用相关工艺数据块和 CPU 的专用 PTO（Pulse Train Outputs，脉冲串输出）来控制轴上的运动。 有关运行运动控制指令的信息，请参见 STEP 7 Basic 的在线信息系统。



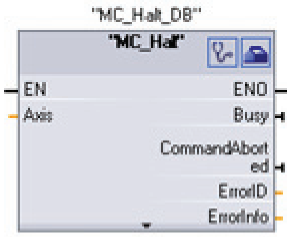
MC_Power 可启用和禁用运动控制轴。



MC_Reset 可复位所有运动控制错误。 所有可确认的运动控制错误都会被确认。



MC_Home 可建立轴控制程序与轴机械定位系统之间的关系。



MC_Halt 可取消所有运动过程并使轴停止运动。 停止位置未定义。



MC_MoveJog 可执行用于进行测试和启动的点动模式。



MC_MoveAbsolute 可启动到某个绝对位置的运动。 该作业在到达目标位置时结束。



MC_MoveRelative 可启动相对于起始位置的定位运动。



MC_MoveVelocity 可使轴以指定的速度行进。

5.4 便于编程的其它功能

5.4.1 系统存储器和时钟存储器提供了标准功能

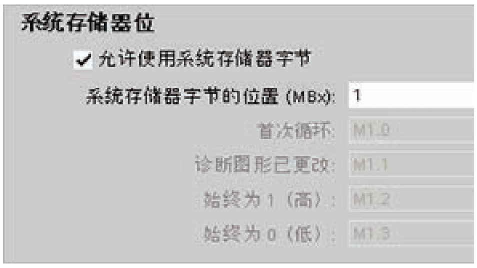
使用 CPU 属性可启用“系统存储器”和“时钟存储器”的相应字节。 程序逻辑可以引用这些功能的各个位。

- 可以将 M 存储器的一个字节分配给系统存储器。 系统存储器的字节提供了以下可供用户程序引用的四位：
  - “始终禁用”位始终设置为 0。
  - “始终启用”位始终设置为 1。
  - “诊断图形已更改”位在 CPU 记录了诊断事件后的一个扫描周期内设置为 1。

    在首次执行程序循环 OB 结束后，CPU 才会设置“诊断图形已更改”位。 因此，在启动 OB 执行期间或首次程序循环 OB 执行期间，用户程序都无法检测到是否出现了诊断更改。
  - “首次扫描”位在启动 OB 完成后的第一次扫描期间设置为 1。 （执行完第一次扫描后，“首次扫描”位将设置为 0。）
- 可以将 M 存储器的一个字节分配给时钟存储器。 被组态为时钟存储器的字节中的每一位都可生成方波脉冲。 时钟存储器字节提供了 8 种不同的频率，其范围从 0.5 Hz（慢）到 10 Hz（快）。 这些位可用作控制位（尤其在与沿指令结合使用时），用于在用户程序中周期性触发动作。

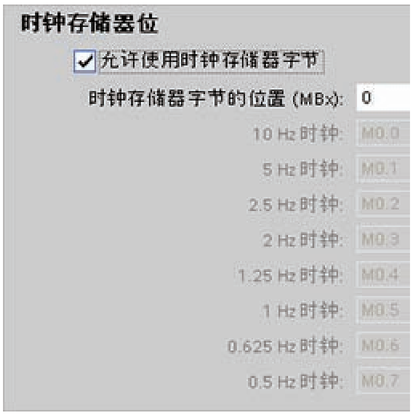
CPU 是在从 STOP 模式切换到 STARTUP 模式时初始化这些字节，并且，在 STARTUP 和 RUN 模式期间，时钟存储器的位随 CPU 时钟同步变化。

因为时钟存储器和系统存储器都是非保留 M 存储器，所以指令或通信可以写入这些单元并破坏其中的数据。 改写系统存储器或时钟存储器字节可能会破坏这些功能中的数据，还可能导致用户程序错误运行。 总是针对用户程序的其它元素不访问的存储器地址，对系统存储器和时钟存储器进行组态。



在以下条件下，系统存储器字节会启用相应位（值 = 1）：

- 首次扫描 (First scan)： 在上电循环后的第一个扫描周期内启用
- 诊断图形已更改 (Diagnostic graph changed)。
- 始终为 1（高）(Always 1 (high))： 始终启用
- 始终为 0（低）(Always 0 (low))： 始终禁用



时钟存储器字节的各个位分别按固定的时间间隔循环启用和禁用。

每个时钟标志都在相应的 M 存储器位产生一个方波脉冲。 这些位可用作控制位（尤其在与沿指令结合使用时），用于在用户代码中周期性触发动作。

说明

例如，当为系统存储器或时钟存储器的位分配 PLC 变量名称时， 为便于引用，变量名称可以描述位的功能，并且用户可以在用户程序中方便地输入变量名称。

5.4.2 监视表格便于监视用户程序

使用“监视表格”监视和修改正在由在线 CPU 执行的用户程序的值。可在项目中创建并保存不同的监视表格以支持各种测试环境。这使得用户可以在调试期间或出于维修和维护目的重新进行测试。



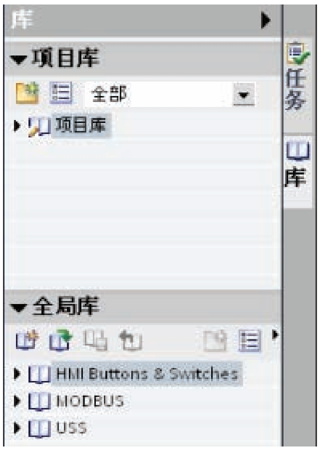
	名称	地址	显示格式	监视值	使用触发器监视	使用触发器修改
1	"Start"	%I0.0	布尔型		永久	永久
2	"Stop"	%I0.1	布尔型		永久	永久
3	"Running"	%I0.2	布尔型		永久	永久
4	"Del"	%I0.3	布尔型		永久	永久

通过监视表格，可监视 CPU 并与 CPU 交互，如同 CPU 执行用户程序一样。 不仅可以显示或更改代码块和数据块的变量值，也可以显示或更改 CPU 存储区（其中包括 输入和输出（I 和 Q）、外围设备输入、位存储器 (M) 和 DB (页 31)）的值。 通过监视表格，可在 STOP 模式下启用 CPU 的外围设备输出（例如“Stop:P”或“Q3.4:P”）。例如，测试 CPU 的接线时可为输出端赋特定值。

监视表格也可用于 “强制”变量或将变量设置为特定值 (页 95)。每次扫描应用一次强制值。 强制值可在程序执行期间更改，但是对于输出，强制值将在扫描结束时写入。 要强制输入或输出（使用“:P”），单击其中一个“强制”(Force) 按钮即可。

5.4.3 便于访问的项目库和全局库

通过全局库和项目库，可以在整个项目中或者在项目间重复使用所存储的对象。 例如，可以创建块模板以便在不同项目中使用并根据自动化任务的特定要求对其进行修改。 可以在这些库中存储各种对象，例如 FC、FB、DB、设备配置、数据类型、监视表格、过程画面和面板。 还可以将 HMI 设备的组件保存在项目中。



每个项目都有一个项目库，用于存储要在项目中多次使用的对象。 该项目库是项目的一部分。 打开或关闭项目时，会相应地打开或关闭项目库；而保存项目时，也会相应地保存项目库中所做的任何更改。

用户可以创建自己的全局库，用于存储供其它项目使用的对象。 创建新的全局库后，可将该库保存在计算机或网络中的某个位置。

STEP 7 Basic 提供了几个可供所有项目使用的全局库。

说明

保存项目时并不会保存或更新全局库。 要保存被添加或修改的全局库，需使用全局库工具栏中的“保存对库所做的更改”(Save the changes to the library) 按钮。

5.4.4 用于显示使用情况的交叉引用

巡视窗口可显示有关所选对象在整个项目中使用情况的交叉引用信息，例如用户程序、CPU 以及任何 HMI 设备。 “交叉引用”(Cross-reference) 选项卡显示使用了所选对象的实例和使用该对象的其它对象。 巡视窗口还包括交叉引用中仅在线可用的块。 要显示交叉引用，请选择“显示交叉引用”(Show cross-references) 命令。 （在项目视图中，可在“工具”(Tools) 菜单中找到交叉引用。）

说明

不必关闭编辑器即可看到交叉引用信息。

可以对交叉引用中的条目进行排序。 交叉引用列表提供用户程序中存储器地址和变量的使用概况。

- 创建和更改程序时，用户始终能够掌握所使用的操作数、变量和块调用情况。
- 从交叉引用可直接跳转到操作数和变量的使用位置。
- 在程序测试或故障排除期间，系统会通知您哪个块中的哪条命令在处理哪个存储单元、哪个画面在使用哪个变量，以及哪个块被其它哪个块调用。

列	说明
对象 (Object)	使用下级对象或被下级对象使用的对象的名称
数量 (Quantity)	使用数量
位置 (Location)	每个使用位置，例如，程序段
属性 (Property)	被引用对象的特定属性，例如，多重背景声明中的变量名称
作为 (as)	显示对象的更多相关信息，例如，背景数据块用作模板还是用作多重背景
访问 (Access)	访问类型，对操作数的访问是读访问 (R)、写访问 (W) 还是二者的组合。
地址 (Address)	操作数的地址
类型 (Type)	有关创建对象所使用的类型和语言的信息
路径 (Path)	对象在项目树中的路径

5.4.5 用于检查调用层级的调用结构

调用结构描述了用户程序中块的调用层级。 其提供了以下几个方面的概要信息：所用的块、对其它块的调用、各个块之间的关系、每个块的数据要求以及块的状态。 可从调用结构打开程序编辑器并对块进行编辑。

显示调用结构时会显示用户程序中使用的块的列表。 STEP 7 Basic 高亮显示调用结构的第一级，并显示未被程序中的其它任何块调用的所有块。 调用结构的第一级显示 OB 以及未被 OB 调用的所有 FC、FB 和 DB。如果某个代码块调用了其它块，则被调用块将以缩进方式显示在调用块的下方。 调用结构仅显示被代码块调用的那些块。

可以选择在调用结构中仅显示导致冲突的块。 下列情况会导致冲突：

- 块执行的任何调用具有更旧或更新代码时间戳
- 块所调用块的接口已更改
- 块所使用变量的地址和/或数据类型已更改
- 块未被 OB 直接或间接调用
- 块调用了不存在的块或缺失的块

可以将多个块调用和数据块分为一组。 可使用下拉列表来查看指向各个调用位置的链接。

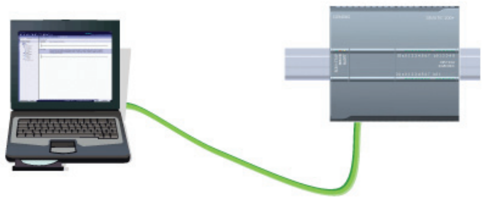
还可执行一致性检查以显示时间戳冲突。 若在生成程序期间或之后更改块的时间戳，将导致时间戳冲突，而这又会导致调用块和被调用块间出现不一致。

- 通过重新编译代码块可纠正大多数时间戳和接口冲突。
- 如果通过编译无法解决不一致问题，可使用“详细资料”(Details) 列中的链接转到程序编辑器中的问题源。 然后可手动消除任何不一致情况。
- 必须重新编译所有以红色标记的块。

轻松实现设备间通信　6

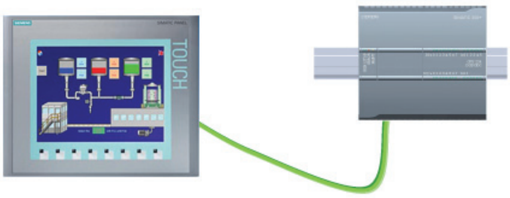
为了能够与下列设备进行通信，CPU 的集成 PROFINET 端口同时支持以太网通信标准和基于 TCP/IP 的通信标准：

- 安装了 STEP 7 Basic 的编程设备
- HMI 设备
- 其它 CPU 或使用标准 TCP 通信协议传输块 (T-block) 指令的非西门子设备



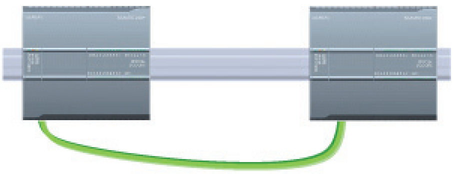
对于 CPU 和编程设备之间的直接连接：

- 项目必须包含 CPU。
- 编程设备不是项目的一部分，但是其必须运行 STEP 7 Basic。



对于 CPU 和 HMI 面板之间的直接连接：

- 项目必须同时包含 CPU 和 HMI。



对于 CPU 和 CPU 之间的直接连接：

- 项目必须同时包含这两个 CPU。
- 必须组态这两个 CPU 之间的网络连接。

对于有两个以上的设备连接在一起的网络：

- 项目必须包含这些设备（CPU 和 HMI）。 不要将路由器包含在组态中。
- 必须组态这些设备之间的网络连接。



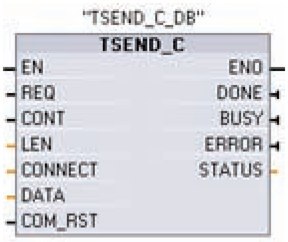
CPU 使用传输连接协议 (TCP, Transport Connection Protocol) 和基于 TCP 的 ISO 传输 (RFC 1006) 应用协议。 组态与 CPU 的 ISO-over-TCP 连接时，请在被动通信伙伴的 TSAP 扩展中仅使用 ASCII 字符。

编程设备或 HMI 与 CPU 之间的直接连接不需要以太网交换机。 含有两个以上的 CPU 或 HMI 设备的网络需要以太网交换机。

说明

CPU 上的 PROFINET 端口不包含以太网交换设备。 可以使用安装在机架上的 Siemens CSM1277 4 端口以太网交换机 ① 来连接 CPU 和 HMI 设备。

6.1 PROFINET 指令（传输块 (T-block)）



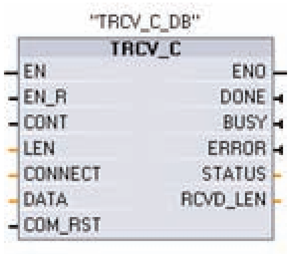
TSEND_C 可与伙伴站建立 TCP 或 ISO on TCP 通信连接，发送数据并且可以终止该连接。 设置并建立连接后，CPU 会自动保持和监视该连接。 TSEND_C 兼具 TCON、TDISCON 和 TSEND 的功能。

请仅在程序循环 OB（如 OB 1）中使用传输块 (T-block) 指令。

- 要建立连接，请在 CONT = 1 时执行 TSEND_C。成功建立连接后，将参数 DONE 置位一个周期。
- 要终止通信连接，请在 CONT = 0 时执行 TSEND_C。连接将立即中止。 这还会影响接收站。 将在接收站关闭该连接，并且接收缓冲区内的数据可能会丢失。
- 要通过建立的连接发送数据，请在 REQ 的上升沿执行 TSEND_C。 发送操作成功执行后，TSEND_C 便会置位 DONE 参数一个周期。
- 要建立连接并发送数据，请在 CONT =1 且 REQ = 1 时执行 TSEND_C。发送操作成功执行后，TSEND_C 便会置位 DONE 参数一个周期。

说明

由于 TSEND_C 采用异步处理，所以在 DONE 参数值或 ERROR 参数值为 TRUE 前，必须保持发送方区域中的数据一致。 对于 TSEND_C，DONE 参数状态为 TRUE 表示数据成功发送。 但并不表示连接伙伴 CPU 实际读取了接收缓冲区。 由于 TRCV_C 采用异步处理，因此仅当参数 DONE = 1 时，接收方区域中的数据才一致。



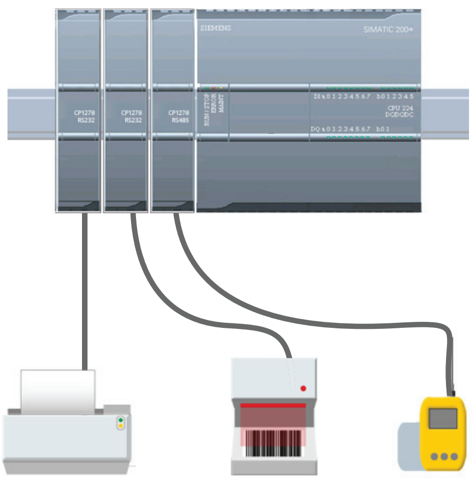
TRCV_C 可与伙伴 CPU 建立 TCP 或 ISO-on-TCP 通信连接，可接收数据，并且可以终止该连接。设置并建立连接后，CPU 会自动保持和监视该连接。 TRCV_C 指令兼具 TCON、TDISCON 和 TRCV 指令的功能。

- 要建立连接，请在参数 CONT = 1 时执行 TRCV_C。
- 要接收数据，请在参数 EN_R = 1 时执行 TRCV_C。参数 EN_R = 1 且 CONT = 1 时可连续接收数据。
- 要终止连接，请在参数 CONT = 0 时执行 TRCV_C。连接将立即中止且数据可能丢失。

说明

处理 TSEND_C 和 TRCV_C 指令花费的时间量无法确定。 要确保这些指令在每次扫描循环中都被处理，务必从主程序循环扫描中对其调用，例如，从程序循环 OB 中或从程序循环扫描中调用的代码块中对其调用。 **不要**从硬件中断 OB、延时中断 OB、循环中断 OB、错误中断 OB 或启动 OB 调用这些指令。

6.2 PtP、USS 和 Modbus 通信协议

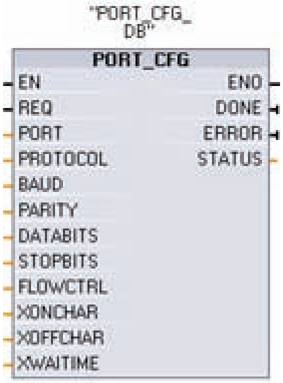


CPU 支持使用 PtP 协议进行基于字符的串行通信，在该通信中，用户应用程序可完全定义和实施所选的协议。PtP 使得众多可能得以实现：

- 将信息直接发送到外部设备，例如打印机
- 从其它设备（例如，条码阅读器、RFID 阅读器、第三方照相机或视觉系统以及许多其它类型的设备）接收信息
- 与其它设备（例如，GPS 设备、第三方照相机或视觉系统，或者无线调制解调器）进行数据的发送和接收

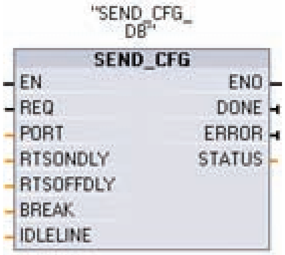
PtP 是一种支持各种波特率和奇偶校验选项的串行通信。 STEP 7 Basic 提供了指令库，可用来针对您的应用进行编程。 这些库针对 USS 驱动协议（仅限 RS485）以及 Modbus RTU 主站和 RTU 从站协议提供了 PtP 通信功能。

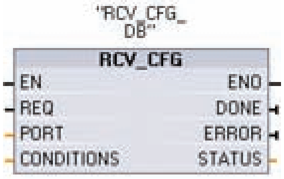
6.2.1 PtP 指令



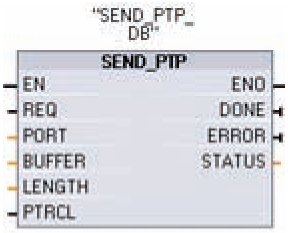
通过 PORT_CFG、SEND_CFG 和 RCV_CFG 指令，可以从用户程序中更改组态。

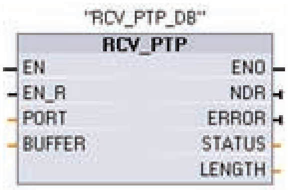
- PORT_CFG 用于更改端口参数，例如波特率。
- SEND_CFG 用于更改串行传输参数的组态。
- RCV_CFG 用于更改通信端口中串行接收方参数的组态。该指令可组态表示接收消息开始和结束的条件。RCV_PTP 指令将接收满足这些条件的消息。





动态组态更改不会永久存储在 CPU 中。 在上电循环后，将使用设备配置中的初始静态组态。





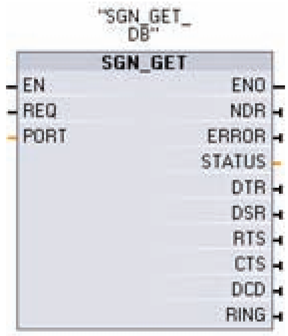
SEND_PTP、RCV_PTP 和 RCV_RST 指令用于控制 PtP 通信：

- SEND_PTP 可将指定的缓冲区数据传送到 CM 模块。 在模块以指定波特率发送数据的同时，CPU 会继续执行用户程序。
- RCV_PTP 可检查 CM 模块中已接收的消息。 如果有消息，则会将其从 CM 传送到 CPU。
- RCV_RST 可复位接收缓冲区。

每个 CM 模块最多可以缓冲 1K 字节。可以跨多个接收的消息来分配该缓冲区。



SGN_SET 和 SGN_GET 仅对 RS232 CM 模块有效。 这些指令用于读取或设置 RS232 通信信号。



6.2.2 USS 指令库

USS 库支持 USS 协议，并提供了专门用于通过 CM 模块的 RS485 端口与驱动器进行通信的功能。 可使用 USS 库控制物理驱动器和读/写驱动器参数。 每个 RS485 CM 最多可支持 16 个驱动器。

- USS_PORT 指令处理 CPU 和连接到同一个 CM 的所有驱动器之间的实际通信。请在应用程序中分别为每个 CM 插入一个不同的 USS_PORT 指令。 请确保用户程序以足够快的速度执行 USS_PORT 指令，以防止与驱动器通信超时。 USS_PORT 指令用于程序循环 OB 或任何中断 OB。
- USS_DRV 指令用于访问 USS 网络中指定的驱动器。 USS_DRV 指令的输入和输出参数代表驱动器的状态和控制。 如果网络上有 16 个驱动器，则用户程序必须至少具有 16 个 USS_DRV 指令，每个驱动器一个指令。 请确保 CPU 以控制驱动器功能所需的速率执行 USS_DRV 指令。 USS_DRV 指令只能用于程序循环 OB。
- USS_RPM 和 USS_WPM 指令用于读取和写入远程驱动器的工作参数。 这些参数控制驱动器的内部运行。 有关这些参数的定义，请参见驱动器手册。 用户程序可根据需要包含任意数量的此类指令。 但在任何特定时刻，每个驱动器只能激活一个读或写请求。 USS_RPM 和 USS_WPM 指令只能用于程序循环 OB。

对于与各个 CM 模块相连接的 USS 网络，背景数据块包含用于该网络中所有驱动器的临时存储区和缓冲区。 驱动器的 USS 指令通过背景数据块来共享信息。



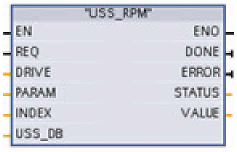
USS_DRV 指令通过创建请求消息和解释驱动器响应消息与驱动器交换数据。 与某个 USS 网络和 CM 相关联的所有 USS 指令必须使用相同的背景数据块。

每个驱动器使用一个单独的 USS_DRV 指令。

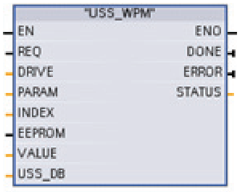


USS_PORT 指令用于处理 USS 网络上的通信。 通常，每个 CM 只使用一个 USS_PORT 指令，USS_PORT 指令用于处理与单个驱动器之间的通信。

在延时中断 OB 中执行 USS_PORT 以防止驱动器超时，并使 USS_DRV 调用可以使用最新的 USS 更新数据。



USS_RPM 指令用于从驱动器读取参数。 在程序循环 OB 中执行 USS_RPM。



USS_WPM 指令用于修改驱动器中的参数。 在程序循环 OB 中执行 USS_WPM。

参数“EEPROM”用于控制将数据写入到 EEPROM 中。 要延长 EEPROM 的使用寿命，请使用参数“EEPROM”将 EEPROM 写操作的次数降到最低。

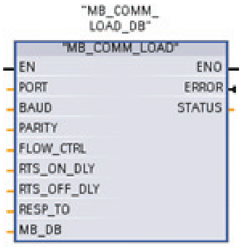
计算与驱动器通信所需的时间

与驱动器进行的通信与 CPU 扫描不同步。 在完成一个驱动器通信事务之前，CPU 通常完成了多次扫描。

USS_PORT 间隔是一个驱动器事务所需的时间。 下表列出了各个波特率对应的最小 USS_PORT 间隔。USS_PORT 功能的调用间隔比 USS_PORT 间隔短时，并不会增加事务数。如果通信错误导致尝试 3 次才能完成事务，则驱动器超时间隔是处理该事务可能花费的时间。默认情况下，USS 协议库对每个事务最多自动进行 2 次重试。

波特率	计算的最小 USS_PORT 调用间隔（毫秒）
1200	790
2400	405
4800	212.5
9600	116.3
19200	68.2
38400	44.1
57600	36.1
115200	28.1

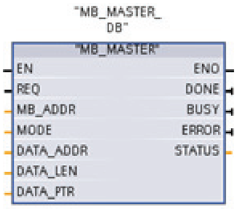
6.2.3 Modbus 指令库



MB_COMM_LOAD 指令用于组态 CM 模块上的端口，以进行 Modbus RTU 协议通信。

可以使用 RS232 或 RS485 CM 模块。

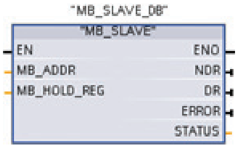
用户程序必须先执行 MB_COMM_LOAD 来组态端口，之后 MB_SLAVE 或 MB_MASTER 指令才能与该端口进行通信。



MB_MASTER 指令允许用户程序作为 Modbus 主站进行通信。可访问一个或多个 Modbus 从站设备中的数据。

插入 MB_MASTER 指令可创建背景数据块。 使用此 DB 名称作为 MB_COMM_LOAD 指令中的 MB_DB 参数。

从同一个 OB（或 OB 优先等级）执行指定端口的所有 MB_MASTER 指令。



MB_SLAVE 指令允许用户程序作为 Modbus 从站进行通信。Modbus RTU 主站可以发出请求，然后程序通过执行 MB_SLAVE 来响应。

插入 MB_SLAVE 指令可创建背景数据块。 使用此 DB 名称作为 MB_COMM_LOAD 指令中的 MB_DB 参数。

从循环中断 OB 执行所有的 MB_SLAVE 指令。

Modbus 指令不使用通信中断事件来控制通信过程。用户程序必须轮询 MB_MASTER 或 MB_SLAVE 指令以了解传送和接收的完成情况。

如果某个端口作为从站响应 Modbus 主站，则 MB_MASTER 无法使用该端口。对于给定端口，只能使用一个 MB_SLAVE 执行实例。 同样，如果要将某个端口用于初始化 Modbus 主站的请求，则 MB_SLAVE 将不能使用该端口。MB_MASTER 执行的一个或多个实例可使用该端口。

如果用户程序操作 Modbus 从站，则对 MB_SLAVE 的轮询（周期性执行）速度有要求，即必须使该指令能及时响应来自 Modbus 主站的进入请求。

如果用户程序操作 Modbus 主站并使用 MB_MASTER 向从站发送请求，则用户必须继续轮询（执行 MB_MASTER）直到返回从站的响应。

7 轻松使用内置脉冲发生器



可以组态 CPU 或信号板 (SB, Signal Board) 的输出，使其具有脉冲发生器或脉冲串输出 (PTO, Pulse-Train Output) 的功能。 脉冲宽度调制 (PWM, Pulse-Width Modulation) 指令和基本运动指令使用这些输出。

有关基本运动指令的信息，请参见 STEP 7 Basic 的在线帮助。

说明

用户程序中的其它指令无法使用脉冲串输出

将 CPU 或 SB 的输出组态为脉冲发生器时（供 PWM 或基本运动控制指令使用），这会从 Q 存储器中移除相应的输出地址（Q0.0、Q0.1、Q4.0 和 Q4.1），并且这些地址在用户程序中不能用于其它用途。 如果用户程序向用作脉冲发生器的输出写入值，则 CPU 不会将该值写入到物理输出。

注意
请勿超出最大脉冲频率。 如 S7-1200 系统手册中所述，脉冲输出发生器的最大脉冲频率对于 CPU 的数字量输出为 100 KHz，而对于信号板的数字量输出为 20 KHz。 组态基本运动指令时，请注意，如果被组态轴的最大速度或频率超出了此硬件限制，STEP 7 Basic 并不会提醒用户。 这可能会导致应用出现问题，因此请始终确保不会超出硬件的最大脉冲频率。

7.1 高速计数器

高速计数器 (HSC, High-Speed Counter) 可用作增量轴编码器的输入。 该轴编码器每转提供指定数量的计数值以及一个复位脉冲。 来自轴编码器的时钟和复位脉冲将输入到 HSC 中。

先是将若干预设值中的第一个装载到 HSC 上，并且在当前计数值小于当前预设值的时段内计数器输出一直是激活的。 在当前计数值等于预设值时、发生复位时以及方向改变时，HSC 会提供一个中断。

每次出现“当前计数值等于预设值”中断事件时，都将装载一个新的预设值，同时会设置输出的下一状态。 当出现复位中断事件时，将设置输出的第一个预设值和第一个输出状态，并重复该循环。

由于中断发生的频率远低于 HSC 的计数速率，因此能够在对 CPU 扫描周期影响相对较小的情况下实现对高速操作的精确控制。 通过提供中断，可以在独立的中断例程中执行每次的新预设值装载操作以实现简单的状态控制。 （或者，也可在单个中断例程中处理所有中断事件。）

选择 HSC 的功能

所有 HSC 在同一计数器运行模式下的工作方式都相同。 HSC 共有四种基本类型： 具有内部方向控制的单相计数器、具有外部方向控制的单相计数器、具有 2 个时钟输入的双相计数器以及 A/B 相正交计数器。 请注意，每个计数器并非支持每一种模式。 用户可选择是否激活复位输入来使用各种 HSC 类型。 如果激活复位输入，则它会清除当前值并在您禁用复位之前保持清除状态。

频率功能： 有些 HSC 模式允许 HSC 被组态（计数类型）为报告频率而非当前脉冲计数值。 有三种可用的频率测量周期： 0.01、0.1 或 1.0 秒。

频率测量周期决定 HSC 计算并报告新频率值的频率。 报告频率是通过上一测量周期内总计数值确定的平均值。 如果该频率在快速变化，则报告值将是介于测量周期内出现的最高频率和最低频率之间的一个中间值。 无论频率测量周期的设置是什么，总是会以赫兹为单位来报告频率（每秒脉冲个数）。

计数器的模式和输入： 下表列出了用于与 HSC 相关的时钟、方向控制和复位功能的输入。 同一输入不可用于两个不同的功能，但任何未被其 HSC 的当前模式使用的输入均可用于其它用途。

例如，如果 HSC1 处于使用内置输入但未使用外部复位 (I0.3) 的模式下，则 I0.3 可以用于沿中断或 HSC2。

<table>
<tr><th colspan="3">说明</th><th colspan="3">默认输入分配</th><th>功能</th></tr>
<tr><td rowspan="6">HSC</td><td>HSC1</td><td>内置
或信号板
或监视 PTO 0[1]</td><td>I0.0
I4.0
PTO 0 脉冲</td><td>I0.1
I4.1
PTO 0 方向</td><td>I0.3
I4.3
-</td><td rowspan="6"></td></tr>
<tr><td>HSC2</td><td>内置
或信号板
或监视 PTO 1[1]</td><td>I0.2
I4.2
PTO 1 脉冲</td><td>I0.3
I4.3
PTO 1 方向</td><td>I0.1
I4.1
-</td></tr>
<tr><td>HSC3[2]</td><td>内置</td><td>I0.4</td><td>I0.5</td><td>I0.7</td></tr>
<tr><td>HSC4[3]</td><td>内置</td><td>I0.6</td><td>I0.7</td><td>I0.5</td></tr>
<tr><td>HSC5[4]</td><td>内置
或信号板</td><td>I1.0
I4.0</td><td>I1.1
I4.1</td><td>I1.2
I4.3</td></tr>
<tr><td>HSC6[4]</td><td>内置
或信号板</td><td>I1.3
I4.2</td><td>I1.4
I4.3</td><td>I1.5
I4.1</td></tr>
<tr><td rowspan="9">模式</td><td colspan="2" rowspan="2">具有内部方向控制的单相计数器</td><td rowspan="2">时钟</td><td rowspan="2">-</td><td>-</td><td>计数或频率</td></tr>
<tr><td>复位</td><td>计数</td></tr>
<tr><td colspan="2" rowspan="2">具有外部方向控制的单相计数器</td><td rowspan="2">时钟</td><td rowspan="2">方向</td><td>-</td><td>计数或频率</td></tr>
<tr><td>复位</td><td>计数</td></tr>
<tr><td colspan="2" rowspan="2">具有 2 个时钟输入的双相计数器</td><td rowspan="2">加时钟</td><td rowspan="2">减时钟</td><td>-</td><td>计数或频率</td></tr>
<tr><td>复位</td><td>计数</td></tr>
<tr><td colspan="2" rowspan="2">A/B 相正交计数器</td><td rowspan="2">A 相</td><td rowspan="2">B 相</td><td>-</td><td>计数或频率</td></tr>
<tr><td>Z 相</td><td>计数</td></tr>
<tr><td colspan="2">监视脉冲串输出 (PTO)[1]</td><td>时钟</td><td>方向</td><td>-</td><td>计数</td></tr>
</table>

[1] 脉冲串输出监视功能始终使用时钟和方向。 如果仅为脉冲组态了相应的 PTO 输出，则通常应将方向输出设置为正计数。

[2] 对于仅支持 6 个内置输入的 CPU 1211C，不能使用带复位输入的 HSC3。

[3] 对于仅支持 6 个内置输入的 CPU 1211C，不能使用 HSC4。

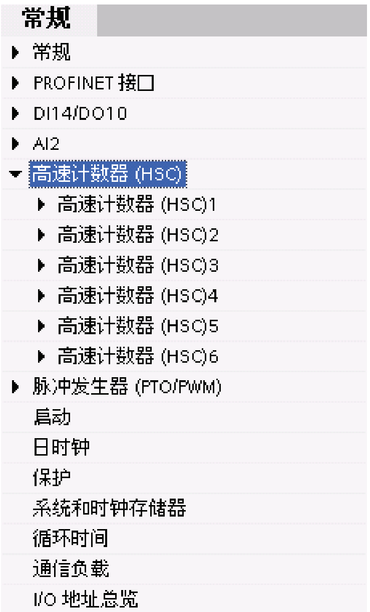
[4] 仅当安装信号板时，CPU 1211C 和 CPU 1212C 才支持 HSC5 和 HSC6。

访问 HSC 的当前值

CPU 将每个 HSC 的当前值存储在一个输入 (I) 地址中。 下表列出了为每个 HSC 的当前值分配的默认地址。 可以通过修改 CPU 的属性 (页 44)来更改当前值的输入地址。

高速计数器	数据类型	默认地址
HSC1	DInt	ID1000
HSC2	DInt	ID1004
HSC3	DInt	ID1008
HSC4	DInt	ID1012
HSC5	DInt	ID1016
HSC6	DInt	ID1020

组态 HSC

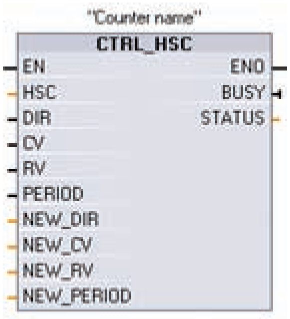


CPU 允许用户组态最多 6 个高速计数器。 用户可编辑 CPU 的“属性”(Properties) 来组态各个 HSC 的参数。

通过编辑 CPU 的“属性”(Properties) 来组态高速计数器的参数，例如，计数器功能、初始值、复位选项和中断事件。

组态 HSC 之后，在用户程序中使用 CTRL_HSC 指令控制 HSC 的运行。

使用 CTRL_HSC 指令



CTRL_HSC 指令可控制用于对发生速率比 CPU 扫描速率更快的事件进行计数的高速计数器。

每个 CTRL_HSC 指令都将数据存储在背景数据块中。 将 CTRL_HSC 指令插入到用户程序中可创建此背景数据块。

参数	数据类型	说明
HSC	HW_HSC	HSC 标识符
DIR	BOOL	1 = 请求新方向
CV	BOOL	1 = 请求设置新的计数器值
RV	BOOL	1 = 请求设置新的参考值
PERIOD	BOOL	1 = 请求设置新的周期值 （仅限频率测量模式）
NEW_DIR	INT	新方向： 1= 向前，-1= 向后
NEW_CV	DINT	新计数器值
NEW_RV	DINT	新参考值
NEW_PERIOD	INT	以秒为单位的新周期值： 0.01、0.1 或 1 （仅限频率测量模式）
BUSY	BOOL	功能忙
STATUS	WORD	执行条件代码

虽然 CTU、CTD 和 CTUD 计数器指令的计数速率受到 CPU 扫描速率的限制，但 HSC 的运行与 CPU 扫描不同步，能够以高达 100 kHz 的计数速率对事件进行计数（适用于 HSC 1、2 或 3 以及板载 CPU 计数输入组态）。

必须先在 CPU 设备配置的项目设置中组态高速计数器，然后才能在程序中使用高速计数器。 HSC 设备配置设置包括选择计数模式、I/O 连接、中断分配以及是作为高速计数器还是设备来测量脉冲频率。 无论是否采用程序控制，均可运行高速计数器。

许多高速计数器组态参数只在项目设备配置中进行设置。 有些高速计数器参数在项目设备配置中初始化，但以后可以通过程序控制进行修改。 CTRL_HSC 指令参数提供了计数过程的程序控制：

- 将计数方向设置为 NEW_DIR 值
- 将当前计数值设置为 NEW_CV 值
- 将参考值设置为 NEW_RV 值
- 将周期值（仅限频率测量模式）设置为 NEW_PERIOD 值

如果执行 CTRL_HSC 指令后以下布尔标记值被设置为 1，则相应的 NEW_xxx 值将装载到计数器。 执行一次 CTRL_HSC 指令可处理多个请求（同时设置多个标记）。

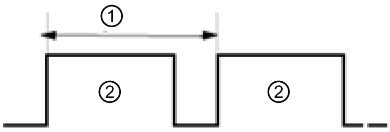
- DIR = 1 表示请求装载 NEW_DIR 值，0 = 无变化
- CV = 1 表示请求装载 NEW_CV 值，0 = 无变化
- RV = 1 表示请求装载 NEW_RV 值，0 = 无变化
- PERIOD = 1 表示请求装载 NEW_PERIOD 值，0 = 无变化

CTRL_HSC 指令通常放置在触发计数器硬件中断事件时执行的硬件中断 OB 中。 例如，如果 CV=RV 事件触发计数器中断，则硬件中断 OB 代码块可以执行 CTRL_HSC 指令以通过装载 NEW_RV 值来更改参考值。

在 CTRL_HSC 参数中没有提供当前计数值。 在高速计数器硬件配置期间分配存储当前计数值的过程映像地址。 可以使用程序逻辑直接读取该计数值，且返回到程序的值将是读取计数器瞬间的正确计数。但计数器仍将继续对高速事件计数。 程序使用旧的计数值完成处理前，实际计数值可能会更改。

7.2 脉冲宽度调制 (PWM)

有两种脉冲发生器可用于控制高速脉冲输出功能： 脉冲宽度调制 (PWM, Pulse Width Modulation) 和脉冲串输出 (PTO, Pulse Train Output)。由于 PWM 输出可在 0 到满刻度之间变化，因此可提供在许多方面都与模拟量输出相同的数字量输出。例如，PWM 输出可用于控制电机的速度，速度范围可以是从停止到全速；也可用于控制阀的位置，位置范围可以是从闭合到完全打开。PTO 由运动控制指令使用。



① 循环时间

② 脉冲宽度时间

占空比可表示为循环时间的百分数或相对量（例如，0 到 1000，或者 0 到 10000）。脉冲宽度可从 0（无脉冲，始终关闭）到满刻度（无脉冲，始终打开）变化。

CTRL_PWM 指令可提供占空比可变的固定循环时间输出。PWM 输出以指定频率（循环时间）启动之后将连续运行。 脉冲宽度会根据需要进行变化以影响控制。

可将每个脉冲发生器指定为 PWM 或 PTO，但不能指定为既是 PWM 又是 PTO。

组态脉冲发生器

这两种脉冲发生器映射到特定的数字量输出，如下表所示。可以使用板载 CPU 输出，也可以使用可选的信号板输出。下表列出了输出点编号（假定使用默认输出组态）。如果更改了输出点编号，则输出点编号将为用户指定的编号。无论是在 CPU 上还是在连接的信号板上，PTO1/PWM1 都使用前两个数字量输出，而 PTO2/PWM2 则使用接下来的两个数字量输出。请注意，PWM 仅需要一个输出，而 PTO 的每个通道可选择使用两个输出。如果脉冲功能不需要输出，则相应的输出可用于其它用途。

说明	默认输出分配	脉冲	方向
PTO 1	板载 CPU	Q0.0	Q0.1
	信号板	Q4.0	Q4.1
PWM 1	板载 CPU	Q0.0	--
	信号板	Q4.0	--
PTO 2	板载 CPU	Q0.2	Q0.3
	信号板	Q4.2	Q4.3
PWM 2	板载 CPU	Q0.2	--
	信号板	Q4.2	--

要准备 PWM 操作，首先通过选择 CPU 来组态设备配置中的脉冲通道，然后组态“脉冲发生器 (PTO/PWM)”，并选择“PWM1”或“PWM2”。启用脉冲发生器（复选框）。如果启用一个脉冲发生器，将为该特定脉冲发生器分配一个唯一的默认名称。 可通过编辑“名称：”(Name:) 编辑框中的名称来更改名称，但必须是唯一的名称。已启用的脉冲发生器的名称将成为“常量”(constant) 变量表中的变量，并可用作 CTRL_PWM 指令的 PWM 参数。可按如下方式重命名脉冲发生器、添加注释以及分配参数：

- 脉冲发生器可用作：PWM 或 PTO（选择 PWM）
- 输出源：板载 CPU 或信号板
- 时间基数：毫秒或微秒
- 脉冲宽度格式：
  - 百分数（0 到 100）
  - 千分数（0 到 1000）
  - 万分数（0 到 10000）
- 循环时间：输入循环时间值。该值只能在此处更改。
- 初始脉冲宽度：输入初始脉冲宽度值。可在运行期间更改脉冲宽度值。
- 起始地址： 输入要在其中查找脉冲宽度值的输出 (Q) 的字长度地址。对于 PWM1，默认位置是 QW1000；而对于 PWM2，默认位置是 QW1002。 该位置的值控制脉冲宽度，并且在每次 PLC 从 STOP 切换到 RUN 模式时都会初始化为上面指定的“初始脉冲宽度：”(Initial pulse width:) 值。在运行期间更改该 Q 字值会引起脉冲宽度变化。

使用 CTRL_PWM 指令



在程序编辑器中放置 CTRL_PWM 指令时，将分配 DB。 CTRL_PWM 指令使用数据块 (DB) 来存储参数信息。 数据块参数由 CTRL_PWM 指令控制。

参数	数据类型	说明
PWM	WORD	PWM 标识符：已启用的脉冲发生器的名称将变为“常量”(constant) 变量表中的变量，并可用作 PWM 参数。
ENABLE	BOOL	1 = 启动脉冲发生器 0 = 停止脉冲发生器
BUSY	BOOL	功能忙
STATUS	WORD	执行条件代码

使用变量名作为 PWM 参数，可指定已启用的脉冲发生器。

输入 EN 为 TRUE 时，PWM_CTRL 指令将根据 ENABLE 输入的值启动或停止所标识的 PWM。脉冲宽度由相关 Q 字输出地址中的值指定。 由于 CPU 在 CTRL_PWM 指令执行后处理请求，所以 S7-1200 CPU 型号的参数 BUSY 总是报告 FALSE。

PLC 第一次进入 RUN 模式时，脉冲宽度将设置为在设备配置中组态的初始值。 根据需要将值写入设备配置中指定的字长度输出 (Q) 地址（“输出地址”/“起始地址：”）以更改脉冲宽度。 使用指令（例如，移动、转换、数学运算）或 PID 将指定的脉冲宽度写入相应的字长度输出 (Q)。 必须使用输出值的有效范围（百分数、千分数、万分数或 S7 模拟格式）。

轻松使用在线工具 8

8.1 转到在线并连接到 CPU

用户通过在线连接可使用以下附加功能：

- 使用 CPU 操作员面板更改 CPU 的工作模式 (页 89)
- 上传、比较和同步用户程序的代码块 (页 90)
- 使用 监视表格 (页 93)测试用户程序及 强制 (页 95)设置 CPU 中的值
- 使用 诊断缓冲区 (页 93)显示事件



要装载项目（其中包括用户程序、设备配置以及 IP 地址），请创建到 CPU 的在线连接。 使用“在线访问”(Online access) 文件夹连接到在线 CPU：

1. 打开“在线访问”(Online access) 文件夹并为 CPU 选择在线连接。
2. 双击“更新可访问的设备”(Update accessible devices) 以显示在线 CPU。

使用“在线工具”(Online tools) 任务卡访问在线 CPU 上的数据。

8.2 将 IP 地址下载到在线 CPU



要分配 IP 地址，必须执行以下任务：

- 组态 CPU (页 46) 的 IP 地址
- 保存组态并将其下载至 CPU。

CPU 的 IP 地址和子网掩码必须与编程设备的 IP 地址和子网掩码兼容。 请咨询网络专家以确定 CPU 的 IP 地址和子网掩码。



如果之前未组态 CPU，也可以使用 “在线访问” (页 87)设置 IP 地址。

已作为设备配置一部分下载的 IP 地址在 PLC 循环上电时不会丢失。

下载包含 IP 地址的设备配置后，用户可以在“在线访问”(Online access) 文件夹下看到该 IP 地址。

8.3 与在线 CPU 交互

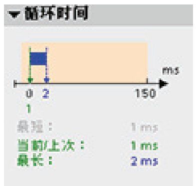
“在线和诊断”(Online and Diagnostics) 门户提供了用于更改在线 CPU 工作模式的操作员面板。 “在线工具”(Online tools) 任务卡显示的操作员面板显示了在线 CPU 的工作模式。也可以通过该操作员面板更改在线 CPU 的工作模式。 使用操作员面板上的按钮更改工作模式（STOP 或 RUN）。 操作员面板还提供了用于复位存储器的 MRES 按钮。



RUN/STOP 指示器的颜色指示 CPU 当前的工作模式： 黄色表示 STOP 模式，而绿色表示 RUN 模式。

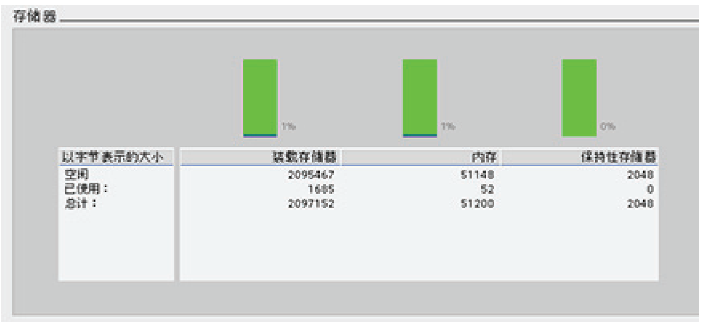
要使用操作员面板，必须在线连接到 CPU。 选择设备配置中的 CPU 或显示在线 CPU 中的代码块之后，可以通过“在线工具”(Online tools) 任务卡显示操作员面板。





用户可以监视在线 CPU 的循环时间。

还可以查看 CPU 的存储器使用情况。





8.4 从在线 CPU 上传

STEP 7 提供了两种从在线 CPU 上传用户程序代码块的方法。



使用项目导航时，可以将代码块从在线 CPU 拖放到离线项目的 CPU 中。

1. 打开项目后，展开“在线访问”(Online access) 容器并选择一个在线 CPU。
2. 展开在线 CPU 以显示用户程序的代码块。
3. 将“程序块”(Program blocks) 文件夹从在线 CPU 拖到离线项目 CPU 的“程序块”(Program blocks) 容器中。

STEP 7 Basic 会将代码块从在线 CPU 复制到离线项目。



还可以使用“比较”功能于在线 CPU 和离线 CPU 之间同步代码块：

1. 选择离线 CPU。
2. 从“工具”(Tools) 菜单中选择“比较离线/在线”(Compare offline/online) 命令。

如果离线 CPU 的代码块与在线 CPU 的代码块不匹配，则可使用“比较”编辑器同步这两个 CPU。



单击“操作”(Action) 图标选择是上传、下载还是不执行任何操作。



单击“同步”(Synchronize) 按钮可将代码块装载到指定的 CPU 或从中装载代码块。

使用“未指定的 CPU”上传硬件配置

如果用户有一个可连接到编程设备的物理 CPU，则很容易上传硬件配置。

用户必须先将 CPU 连接到编程设备，并且必须创建一个新项目。



使用下列两种方法之一来插入“未指定的 CPU”：

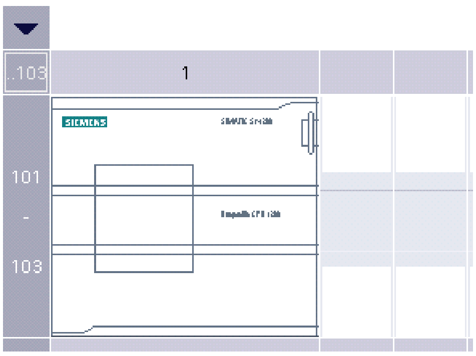
- 在设备配置（项目视图或门户视图）中，添加一个新设备，但要选择“未指定的 CPU”而不是选择特定的 CPU。
- 在门户视图中，单击“新手上路”(First steps) 中的“创建 PLC 程序”(Create a PLC program)。

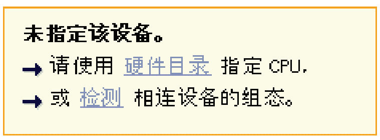
STEP 7 Basic 即会创建一个未指定的 CPU。



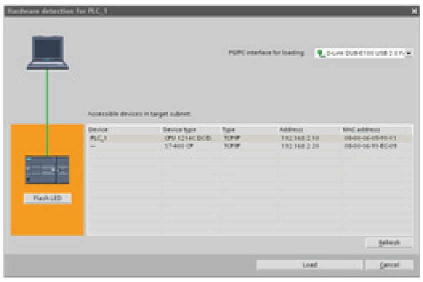
创建未指定的 CPU 后，可以上传在线 CPU 的硬件配置。

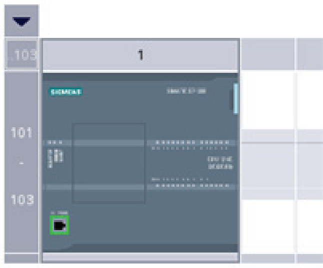
- 在程序编辑器中，从“在线”(Online) 菜单中选择“硬件检测”(Hardware detection) 命令。
- 在设备配置编辑器中，选择用于检测所连设备组态的选项





从在线对话框中选择 CPU 之后，STEP 7 Basic 会上传 CPU 以及所有模块（SM、SB 或 CM）的硬件配置。 不会上传 IP 地址。 必须转到“设备配置”(Device configuration)手动组态 IP 地址。





8.5 比较离线 CPU 与在线 CPU



可以将在线 CPU 中的代码块与离线项目中的代码块进行比较：

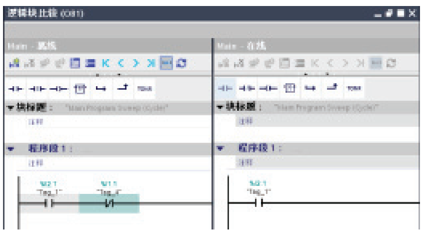
1. 选择离线 CPU。
2. 从“工具”(Tools) 菜单中选择“比较离线/在线”(Compare offline/online) 命令。

如果离线 CPU 的代码块与在线 CPU 的代码块不匹配，则可使用“比较”编辑器同步这两个 CPU。



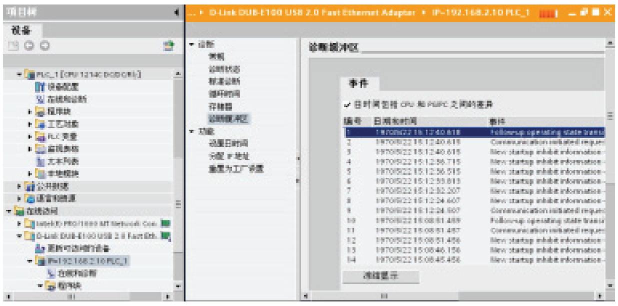
单击“操作”(Action) 图标选择是上传、下载还是不执行任何操作。 

单击“同步”(Synchronize) 按钮可将代码块装载到指定的 CPU。 



单击“详细比较”(detailed comparison) 按钮可并排显示代码块。 详细比较功能会高亮显示在线 CPU 与离线 CPU 的代码块之间的差异。 

8.6 显示诊断事件



CPU 提供了一个诊断缓冲区，其中包含的每个条目对应一个诊断事件，如 CPU 工作模式切换事件，或者 CPU 或模块检测到的错误。

用户必须转到在线才能访问诊断缓冲区。

在 CPU 保持通电时，该日志最多可提供 50 个最新发生的事件。日志填满后，新事件将替换日志中最早的事件。掉电时，将保存最新发生的十个事件。

每个条目都包含了事件发生的日期和时间、事件类别及事件描述。条目按时间顺序显示，最新发生的事件位于最上面。

8.7 使用监视表格监视 CPU

用户通过监视表格可以在 CPU 执行用户程序时监视或修改数据点。 这些数据点可以是输入 (I)、输出 (Q)、外围设备输入或输出（如“On:P”、“I 3.4:P”或“Q3.4:P”）、M 存储器或 DB。 监视功能不会改变程序顺序。它为用户提供有关程序顺序的信息以及 CPU 中的程序的数据。 用户也可以使用“修改”和“强制”功能测试用户程序的执行。

		名称	地址	显示格式	监视值	使用触发器监视	使用触发器修改
1		"Start"	%I0.0	布尔型		永久	永久
2		"Stop"	%I0.1	布尔型		永久	永久
3		"Running"	%I0.2	布尔型		永久	永久
4		"Del"	%I0.3	布尔型		永久	永久

说明

在设备配置期间分配高速计数器 (HSC)、脉冲宽度调制 (PWM) 和脉冲串输出 (PTO) 设备使用的数字 I/O 点。 将数字 I/O 点的地址分配给这些设备之后，无法通过监视表格的“强制”功能修改所分配的 I/O 点的地址值。

使用监视表格可以监视或修改各变量的值。 还可以将变量强制设置为特定值。 用户可以指定在扫描周期开始或结束时、CPU 切换到 STOP 模式时，或者是“永久”（从 STOP 切换到 RUN 后不会重置值）监视或修改变量。



要创建监视表格：

1. 双击“添加新监视表格”(Add new watch table) 打开新监视表格。
2. 输入变量名称将变量添加到监视表格。

要监视变量，必须在线连接到 CPU。可使用以下选项修改变量：

- “立即修改”(Modify now) 立即修改所选地址的值一个扫描周期。
- “使用触发器修改”(Modify with trigger) 修改所选地址的值。

  该功能不提供反馈来指示实际上是否修改了所选地址。如果需要修改反馈，请使用“立即修改”(Modify now) 功能。
- “启用外围设备输出”(Enable peripheral outputs) 允许在 CPU 处于 STOP 模式时启用外围设备输出。 此功能对测试输出模块的接线很有用。

可以使用监视表格顶部的按钮选择各种功能。输入要监视的变量名称并从下拉选择项中选择一种显示格式。在线连接到 CPU 时，单击“监视”(Monitor) 按钮将在“监视值”(Monitor value) 域中显示数据点的实际值。

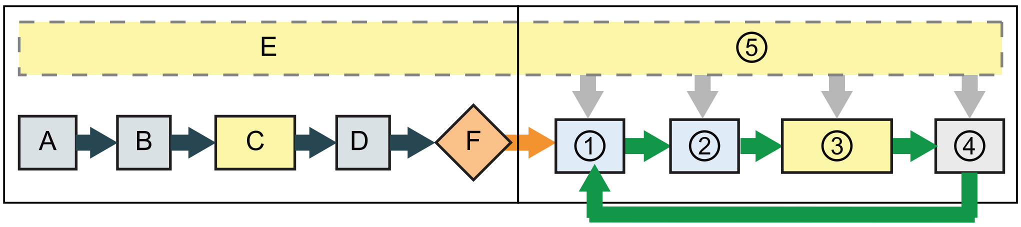
8.8 强制设置 CPU 中的变量

监视表格提供了“强制”功能，能够将与外围设备输入或外围设备输出地址对应的输入或输出点的值改写成特定的值。 CPU 在执行用户程序前将此强制值应用到输入过程映像并在将输出写入到模块前将其应用到输出过程映像。

- 执行扫描之前，CPU 使用强制值改写外围设备输入的值。 用户程序使用该强制值进行处理。
- 扫描结束时，CPU 会将用户程序生成的输出值改写成为外围设备输出指定的任意强制值。强制值出现在物理输出端并被过程使用。

在监视表格中强制输入或输出时，强制操作将变成用户程序的一部分。 如果关闭 STEP 7 Basic，被强制元素对 CPU 正在执行的用户程序而言仍保持激活状态，并直到这些元素被清除。 要清除这些被强制元素，必须使用 STEP 7 Basic 连接到在线 CPU，然后使用监视表格禁用或停止对这些元素进行强制。

如果 CPU 正在执行写保护存储卡上的用户程序，则无法通过监视表格初始化或更改对 I/O 的强制，因为用户无法改写写保护用户程序中的值。 强制写保护值的任何尝试都将生成错误。 如果使用存储卡传送用户程序，则该存储卡上的所有被强制元素都将被传送到 CPU。



启动		RUN	
A	强制功能不影响 I 存储区的清除。	①	将 Q 存储器写入到物理输出期间，CPU 在更新输出时应用强制值。
B	强制功能不影响输出值的初始化。	②	将物理输入的状态复制到 I 存储器后，CPU 应用强制值。
C	启动 OB 执行期间，CPU 在用户程序访问物理输入时应用强制值。	③	用户程序（程序循环 OB）执行期间，CPU 在用户程序访问物理输入时应用强制值。
D	将物理输入的状态复制到 I 存储器后，CPU 应用强制值。	④	自检诊断不受强制功能影响。
E	不影响将中断事件存储到队列。	⑤	不影响在扫描周期的任何时段内处理通信请求和中断。
F	不影响写入到输出的启用。		

技术规范

A

A.1 常规规范

S7-1200 自动化系统符合以下标准和测试规范。 S7-1200 自动化系统的测试标准均基于这些标准和测试规范。

S7-1200 自动化系统满足下列 EC 指令提出的要求和安全相关目标，并且符合欧盟的公报中列出的可编程控制器的协调欧洲标准 (EN)。

- EC 指令 2006/95/EC（低压指令）“设计用于特定电压限值内的电气设备”
  - EN 61131-2:2007 可编程控制器 - 设备要求和测试
- EC 指令 2004/108/EC（EMC 指令）“电磁兼容性”
  - 辐射标准
    EN 61000-6-4:2007： 工业环境
  - 抗扰度标准
    EN 61000-6-2:2005： 工业环境
- EC 指令 94/9/EC (ATEX)“拟用于潜在爆炸性环境的设备和保护系统”
  - EN 60079-15:2005： 保护类型“n”

可向主管部门出具的所持 CE 一致性声明文件位于以下地址：

Siemens AG
IA AS RD ST PLC Amberg
Werner-von-Siemens-Str. 50
D92224 Amberg
Germany

美国保险商实验所，符合

- 美国安全检测实验室公司： UL 508 认证（工业控制设备）
- 加拿大标准协会： CSA C22.2 第 142 号（过程控制设备）

注意
SIMATIC S7-1200 系列符合 CSA 标准。 cULus 标志表示 S7-1200 已通过美国安全检测实验室公司 (UL) 检验和认证，其符合标准 UL 508 和 CSA 22.2 第 142 号。

美国工厂联合研究会 (FM)：
认证标准类别号 3600 和 3611

批准用于：
I 类，2 分区，气体组别 A、B、C、D，温度类别 T4A Ta = 40° C
I 类，2 区，IIC，温度类别 T4 Ta = 40° C

EN 60079-0:2006： 爆炸性环境 - 一般要求

EN 60079-15:2005： 用于潜在爆炸性环境的电气装置；
保护类型“n”

II 3 G Ex nA II T4

要安全使用 S7-1200，必须遵守以下特殊条件：

- 将模块安装在合适的机柜中，根据 EN 60529 至少要提供防护等级 IP54，并且考虑设备将来使用的环境条件。
- 在额定条件下，如果电缆入口点温度超出 70° C 或者导线分支点超出 80° C，则所选电缆的温度规范应符合实际测量温度。
- 应采取措施防止额定电压受暂态干扰而超出 40% 以上。

S7-1200 自动化系统满足 AS/NZS 2064（A 类）标准的要求

海事认证： S7-1200 产品定期向特定机构递交申请以便进行与特定市场和应用有关的认证。 如需要更多有关按零件号排列的最新具体认证列表的信息，请咨询当地西门子代表。

船级社：

- ABS（美国船级社）
- BV（法国船级社）
- DNV（挪威船级社）
- GL（德国船级社）
- LRS（英国劳氏船级社）
- Class NK（日本船级社）

工业环境： S7-1200 自动化系统设计用在工业环境中。

应用现场	噪声排放要求	噪声抗扰度要求
工业	EN 61000-6-4:2007	EN 61000-6-2:2005

电磁兼容性： 电磁兼容性 (EMC) 是电气设备在电磁环境中按预期运行以及运行时电磁干扰的发射水平 (EMI) 不会干扰周围其它电气设备的能力。

电磁兼容性 - 抗扰度符合 EN 61000-6-2	
EN 61000-4-2 静电放电	8 kV，对所有表面的空中放电 6 kV，对暴露导电表面的接触放电
EN 61000-4-3 辐射电磁场	80 到 100 MHz，10 V/m，1 kHz 时 80% AM 1.4 到 2.0 GHz，3 V/m，1 kHz 时 80% AM 2.0 到 2.7 GHz，1 V/m，1 kHz 时 80% AM
EN 61000-4-4 快速瞬变脉冲	2 kV，5 kHz，到 AC 和 DC 系统电源的耦合网络 2 kV，5 kHz，到 I/O 的耦合夹
EN 6100-4-5 浪涌抗扰度	AC 系统 - 2 kV 共模，1kV 差模 DC 系统 - 2 kV 共模，1kV 差模 对于 DC 系统（I/O 信号、DC 电源系统)，需要外部保护。
EN 61000-4-6 传导干扰	150 kHz 到 80 MHz，10 V RMS，1kHz 时 80% AM
EN 61000-4-11 电压骤降	AC 系统 60 Hz 时，0% 持续 1 个周期、40% 持续 12 个周期和 70% 持续 30 个周期

电磁兼容性 - 传导和辐射发射符合 EN 61000-6-4	
传导发射 EN 55011，A 类，组 1 0.15 MHz 到 0.5 MHz 0.5 MHz 到 5 MHz 5 MHz 到 30 MHz	 <79dB (μV) 准峰值；<66 dB (μV) 平均值 <73dB (μV) 准峰值；<60 dB (μV) 平均值 <73dB (μV) 准峰值；<60 dB (μV) 平均值
辐射发射 EN 55011，A 类，组 1 30 MHz 到 230 MHz 230 MHz 到 1 GHz	 <40dB (μV/m) 准峰值；在 10m 处测得 <47dB (μV/m) 准峰值；在 10m 处测得

环境条件

环境条件 - 运输和存储	
EN 60068-2-2，测试 Bb，干热和 EN 60068-2-1，测试 Ab，寒冷	-40° C 到 +70° C
EN 60068-2-30，测试 Db，湿热	25° C 到 55° C，湿度 95%
EN 60068-2-14，测试 Na，温度骤变	-40° C 到 +70° C，停留时间 3 小时，2 个周期
EN 60068-2-32，自由落体	0.3 m，5 次，产品包装
大气压	1080 到 660h Pa（相当于海拔 -1000 到 3500m）

环境条件 - 工作	
环境温度范围 （设备下部 25 mm 进风距离）	0° C 到 55° C，水平安装 0° C 到 45° C，垂直安装 湿度 95%，不结露
大气压	1080 到 795 hPa（相当于海拔 -1000 到 2000m）
污染物浓度	$S0_2$： < 0.5 ppm；H_2S： < 0.1 ppm；RH < 60%，不结露
EN 60068-2-14，测试 Nb，温度变化	5° C 到 55° C，3° C/min
EN 60068-2-27 机械冲击	15 G，11 ms 脉冲，3 个轴向上 6 次冲击
EN 60068-2-6 正弦振动	DIN 导轨安装： 5-9 Hz 时 3.5mm，9 - 150 Hz 时 1G 面板安装： 5 到 9 Hz 时 7.00mm，9 到 150 Hz 时 2G 每个轴 10 次摆动，每分 1 倍频程

高电位绝缘测试	
24 V/5 V 标称电路间 115/230 V 电路对地 115/230 V 电路对 115/230 V 电路 115 V/230V 电路对 24 V/5 V 电路	520 VDC（光隔离边界的型式测试） 1500 VAC 常规测试/1950 VDC 型式测试 1500 VAC 常规测试/1950 VDC 型式测试 1500 VAC 常规测试/3250 VDC 型式测试

保护类别： 保护类别 II 符合 EN 61131-2（不需要保护导线）

防护等级

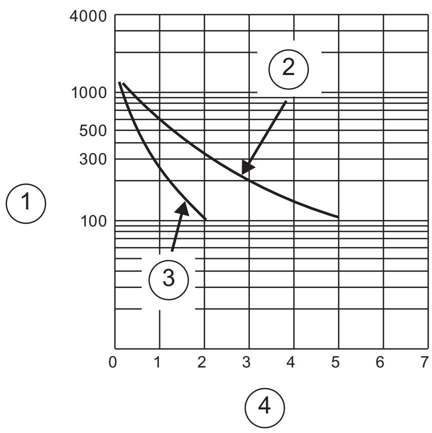
- IP20 机械保护，EN 60529
- 防止手指接触经标准探针测试出的高压。 需要针对灰尘、污物、水和直径小于 12.5mm 的异物施加外部保护。

额定电压

额定电压	容差
24 VDC	20.4 VDC 到 28.8 VDC
120/230 VAC	85 VAC 到 264 VAC，47 到 63 Hz

注意
机械触点接通 S7-1200 CPU 的输出电源或任何数字量信号模块时，会发送信号“1”到数字量输出，时间约 50 微秒。 必须考虑这一点，尤其是使用响应短脉冲的设备时。

继电器电气使用寿命： 继电器供应商提供的典型性能数据如下。 根据具体应用，实际性能可能不同。 使用适合于负载的外部保护电路可增强触点的使用寿命。



① 使用寿命（x 10³ 次动作）

② 250 VAC 阻性负载，
30 VDC 阻性负载

③ 250 VAC 感性负载 (p.f=0.4)
30 VDC 感性负载 (L/R=7ms)

④ 额定工作电流 (A)

A.2 CPU 模块

有关整套技术规范的信息，请参阅 S7-1200 系统手册。

常规规范	CPU 1211C	CPU 1212C	CPU 1214C
尺寸 (W x H x D)	90 x 100 x 75 (mm)	90 x 100 x 75 (mm)	110 x 100 x 75 (mm)
重量 • AC/DC/继电器 • DC/DC/继电器 • DC/DC/DC	 • 420 g • 380 g • 370 g	 • 425 g • 385 g • 370 g	 • 475 g • 435 g • 415 g
功耗 • AC/DC/继电器 • DC/DC/继电器 • DC/DC/DC	 • 10 W • 8 W • 8 W	 • 11 W • 9 W • 9 W	 • 14 W • 12 W • 12 W
SM 和 CM 总线的可用电流 (5 VDC)	最大 750 mA	最大 1000 mA	最大 1600 mA
传感器电源的可用电流 (24 VDC)	最大 300 mA	最大 300 mA	最大 400 mA
数字量输入电流消耗 (24VDC)	所用的每点输入 4 mA	所用的每点输入 4 mA	所用的每点输入 4 mA

CPU 特征	CPU 1211C	CPU 1212C	CPU 1214C
用户存储器 • 工作存储器 • 装载存储器 • 保持存储器	 • 25 KB • 1 MB • 2 KB	 • 25 KB • 1 MB • 2 KB	 • 50 KB • 2 MB • 2 KB
板载数字 I/O	6 点输入 4 点输出	8 点输入 6 点输出	14 点输入 10 点输出
板载模拟 I/O	2 路输入	2 路输入	2 路输入
过程映像大小 • 输入 • 输出	 • 1024 个字节 • 1024 个字节	 • 1024 个字节 • 1024 个字节	 • 1024 个字节 • 1024 个字节
位存储器 (M)	4096 个字节	4096 个字节	8192 个字节

CPU 特征	CPU 1211C	CPU 1212C	CPU 1214C
SM 模块扩展	无	最多 2 个信号模块	最多 8 个信号模块
SB 扩展	最多 1 块信号板	最多 1 块信号板	最多 1 块信号板
CM 扩展	最多 3 个通信模块	最多 3 个通信模块	最多 3 个通信模块
高速计数器 • 单相（时钟频率） • 正交相位（时钟频率）	共 3 个 • 3 个，100 kHz • 3 个，80 kHz	共 4 个 • 3 个，100 kHz 1 个，30 kHz • 3 个，80 kHz 1 个，20 kHz	共 6 个 • 3 个，100 kHz 3 个，30 kHz • 3 个，80 kHz 3 个，20 kHz
脉冲输出	2	2	2
脉冲捕捉输入	6	8	14
延时中断/循环中断	共 4 个 精度为 1 ms	共 4 个 精度为 1 ms	共 4 个 精度为 1 ms
沿中断 使用可选 SB 时	6 个上升沿和 6 个下降沿 10 个上升沿和 10 个下降沿	8 个上升沿和 8 个下降沿 12 个上升沿和 12 个下降沿	12 个上升沿和 12 个下降沿 14 个上升沿和 14 个下降沿
实时时钟 • 精度 • 保持时间（免维护超级电容）	 • +/- 60 秒/月 • 通常为 10 天，40°C 时最少为 6 天	 • +/- 60 秒/月 • 通常为 10 天，40°C 时最少为 6 天	 • +/- 60 秒/月 • 通常为 10 天，40°C 时最少为 6 天
执行速度 • 布尔运算 • 移动字 • 实数数学运算	 • 0.1 μs/指令 • 12 μs/指令 • 18 μs/指令	 • 0.1 μs/指令 • 12 μs/指令 • 18 μs/指令	 • 0.1 μs/指令 • 12 μs/指令 • 18 μs/指令

CPU 特征	CPU 1211C	CPU 1212C	CPU 1214C
通信 • 数据传输率 • 隔离（外部信号与 PLC 逻辑侧） • 电缆类型	1 个以太网端口 • 10/100 Mb/s • 变压器隔离，1500 VDC • CAT5e 屏蔽电缆	1 个以太网端口 • 10/100 Mb/s • 变压器隔离，1500 VDC • CAT5e 屏蔽电缆	1 个以太网端口 • 10/100 Mb/s • 变压器隔离，1500 VDC • CAT5e 屏蔽电缆
连接数 • HMI • PG • 用户程序 • CPU 与 CPU	 • 3 • 1 • 8 • 3	 • 3 • 1 • 8 • 3	 • 3 • 1 • 8 • 3

数字量输入	说明	
输入点数 • CPU 1211C • CPU 1212C • CPU 1214C	总计 • 6 • 8 • 14	可同时使用的点数 • 6 • 8 • 14
类型	漏型/源型（IEC 1 类漏型）	
额定电压	4 mA 时 24 VDC，额定值	
允许的连续电压	最大 30 VDC	
浪涌电压	35 VDC，持续 0.5 s	
逻辑 1 信号（最小） 逻辑 0 信号（最大）	2.5 mA 时 15 VDC 1 mA 时 5 VDC	
隔离（现场侧与逻辑侧）	500 VAC，持续 1 min	
隔离组	1	
滤波时间	0.2、0.4、0.8、1.6、3.2、6.4 和 12.8 ms（可选择，4 个为一组）	

数字量输入	说明
HSC 时钟输入频率（最大） • CPU 1211C • CPU 1212C • CPU 1214C	逻辑 1 电平 = 15 到 26 VDC • 单相： 100 KHz 正交相位： 80 KHz • 单相： 100 KHz（Ia.0 到 Ia.5）和 30 KHz（Ia.6 到 Ia.7） 正交相位： 80 KHz（Ia.0 到 Ia.5）和 20 KHz（Ia.6 到 Ia.7） • 单相： 100 KHz（Ia.0 到 Ia.5）和 30 KHz（Ia.6 到 Ib.5） 正交相位： 80 KHz（Ia.0 到 Ia.5）和 20 KHz（Ia.6 到 Ib.5）
电缆长度（米）	500（屏蔽）；300（非屏蔽）；50（屏蔽，HSC 输入）

数字量输出	继电器	DC
输出点数	AC/DC/继电器和 DC/DC/继电器 • CPU 1211C： 4 • CPU 1212C： 6 • CPU 1214C： 10	DC/DC/DC • CPU 1211C：4 • CPU 1212C：6 • CPU 1214C： 10
可同时使用的输出点数	AC/DC/继电器和 DC/DC/继电器 • CPU 1211C：4 • CPU 1212C：6 • CPU 1214C：10	DC/DC/DC • CPU 1211C：4 • CPU 1212C：6 • CPU 1214C： 10
类型	继电器，干触点	固态 - MOSFET
电压范围	5 到 30 VDC 或 5 到 250 VAC	20.4 到 28.8 VDC
最大电流时的逻辑 1 信号 具有 10 KΩ 负载时的逻辑 0 信号	不适用 不适用	最小 20 VDC 最大 0.1 VDC
电流（最大）	2.0 A	0.5 A
灯负载	30 W DC/200 W AC	5 W
通态电阻	新设备最大为 0.2 Ω	最大 0.6 Ω
每点的漏泄电流	不适用	最大 10 μA
浪涌电流	触点闭合时为 7 A	8 A，最长持续 100 ms
过载保护	无	无

数字量输出	继电器	DC
隔离（现场侧与逻辑侧）	• 线圈与触点，1500 VAC，持续 1 min • 线圈与逻辑侧： 无	500 VAC，持续 1 min
隔离电阻	新设备最小为 100 MΩ	不适用
断开触点间的绝缘	750 VAC，持续 1 min	不适用
隔离组	AC/DC/继电器和 DC/DC/继电器 • CPU 1211C：1 • CPU 1212C：2 • CPU 1214C：2	DC/DC/DC • CPU 1211C：1 • CPU 1212C：1 • CPU 1214C： 1
电感钳位电压	不适用	L+ - 48 VDC，1 W 损耗
开关延迟（Qa.0 到 Qa.3）	最长 10 ms	断开到接通最长为 1.0 μs 接通到断开最长为 3.0 μs
开关延迟（Qa.4 到 Qb.1）	最长 10 ms	断开到接通最长为 50 μs 接通到断开最长为 200 μs
脉冲串输出频率 （Qa.0 和 Qa.2）	不推荐	最大 100 KHz，最小 2 Hz
机械寿命（无负载） 额定负载下的触点寿命	10,000,000 个断开/闭合周期 100,000 个断开/闭合周期	不适用
RUN 到 STOP 时的行为	上一个值或替换值（默认值为 0）	
电缆长度（米）	500 m（屏蔽）；150 m（非屏蔽）	

模拟量输入	说明
输入的路数和类型	2 路电压（单侧）输入 CPU 1211C、CPU 1212C 和 CPU 1214C
范围	0 到 10 V
满量程范围（数据字）[1]	0 到 27648
过冲范围（数据字）[1]	27649 到 32511
溢出（数据字）[1]	32512 到 32767
精度	10 位
最大耐压	35 VDC

模拟量输入	说明
滤波 [2]	无、弱、中或强
噪声抑制 [3]	10、50 或 60 Hz
阻抗	≥100 KΩ
隔离（现场侧与逻辑侧）	无
精度（25°C/0 到 55°C）	满量程的 3.0%/3.5%
共模抑制	40 dB，DC 到 60 Hz
工作信号范围	信号加共模电压必须小于 +12 V 且大于 -12 V
电缆长度	100 m，屏蔽双绞线

1 请参考 S7-1200 系统手册，了解有关模拟量输入的电压和电流表示法。

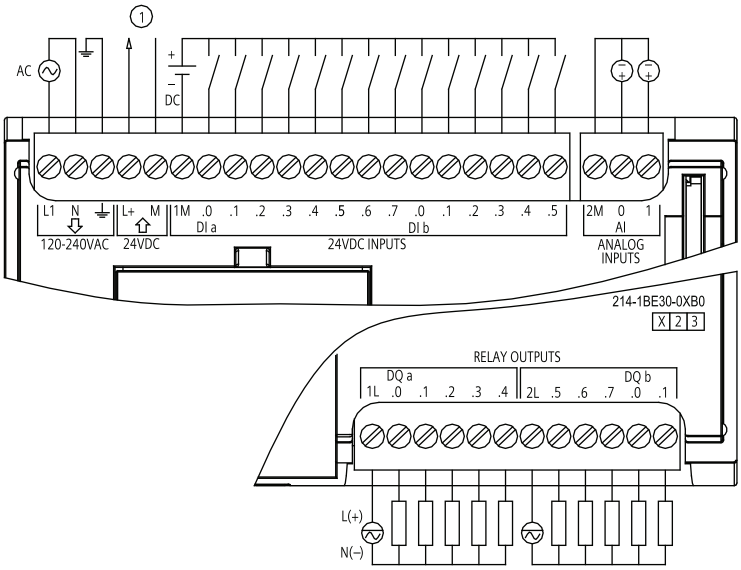
2 请参考 S7-1200 系统手册，了解有关模拟量输入的阶跃响应时间。

3 请参考 S7-1200 系统手册，了解有关模拟量输入的采样速率。

S7-1200 CPU 的接线图示例

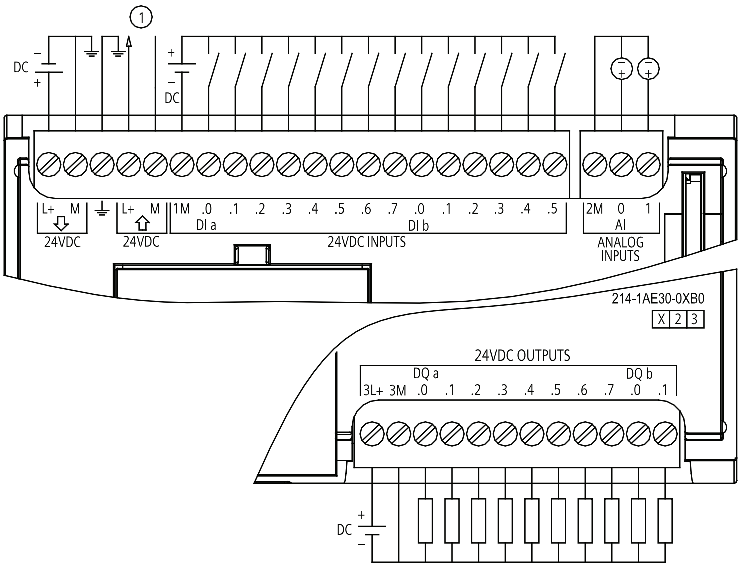
有关详细信息，请参见 S7-1200 系统手册。

CPU 1214C AC/DC/继电器



① 24 VDC 传感器电源输出

CPU 1214C DC/DC/DC



① 24 VDC 传感器电源输出

A.3 信号板

常规	SB 1223 DI 2x24VDC，DQ 2x24VDC	SB 1223 AQ 1x12 位
订货号	6ES7 223-0BD30-0XB0	6ES7 232-4HA30-0XB0
尺寸 (W x H x D)	38 x 62 x 21 (mm)	38 x 62 x 21 (mm)
重量	40 g	40 g
功耗	1.0 W	1.5 W
电流消耗（SM 总线）	50 mA	15 mA
电流消耗 (24 VDC)	所用的每点输入 4 mA	40 mA（无负载）
输入/输出	2 点输入（IEC 1 类漏型） 2 点输出（固态 - MOSFET）	1 点输出 （电压或电流）

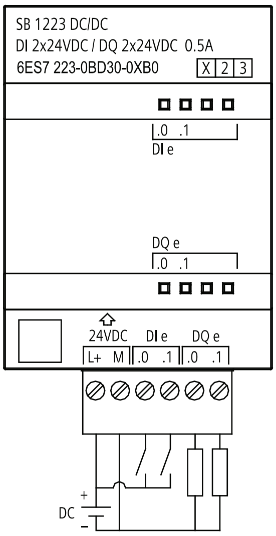
数字量输入	SB 1223 DI 2x24VDC，DQ 2x24VDC
输入的点数和类型 （可同时使用的输入点数）	IEC 1 类漏型： 2 点输入 (2)
额定电压	4 mA 时 24 VDC，额定值
允许的连续电压	最大 30 VDC
浪涌电压	35 VDC，持续 0.5 s
逻辑 1 信号（最小） 逻辑 0 信号（最大）	2.5 mA 时 15 VDC 1 mA 时 5 VDC
HSC 时钟输入频率（最大）	20 kHz（15 到 30 VDC）；30 kHz（15 到 26 VDC）
隔离（现场侧与逻辑侧） 隔离组	500 VAC，持续 1 min 1
滤波时间	0.2、0.4、0.8、1.6、3.2、6.4 和 12.8 ms 可选择，2 个为一组
电缆长度（米）	500（屏蔽）；300（非屏蔽）

数字量输出	SB 1223 DI 2x24VDC，DQ 2x24VDC
输出的点数和类型 （可同时使用的输出点数）	固态 - MOSFET： 2 点输出 (2)
电压范围	20.4 到 28.8 VDC
最大电流时的逻辑 1 信号	最小 20 VDC
具有 10 KΩ 负载时的逻辑 0 信号	最大 0.1 VDC
电流（最大）	0.5 A
灯负载	5 W
通态触点电阻	最大 0.6 Ω
每点的漏泄电流	最大 10 μA
脉冲串输出频率	最大 20 KHz，最小 2 Hz
浪涌电流	5 A，最长持续 100 ms
过载保护	无
隔离（现场侧与逻辑侧） 隔离组	500 VAC，持续 1 min 1
每个公共端的电流	1 A
电感钳位电压	L+ - 48 V，1 W 损耗
开关延迟	断开到接通最长为 2 μs；接通到断开最长为 10 μs
RUN 到 STOP 时的行为	上一个值或替换值（默认值为 0）
电缆长度（米）	500（屏蔽）；150（非屏蔽）

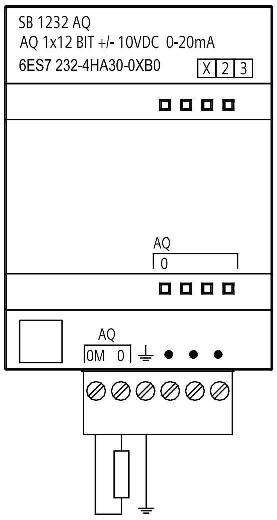
模拟量输出	SB 1223 AQ 1x12 位	
输出的路数和类型	1 路（电压或电流）	
范围	±10 V 或 0 到 20 mA	
精度	电压： 12 位	电流： 11 位
满量程范围（数据字）	电压： -27,648 到 27,648	电流： 0 到 27,648
精度（25°C/0 到 55°C）	满量程的 ±0.5%/±1%	
稳定时间（新值的 95%）	电压： 300 μS (R)，750 μS (1 uF)	电流： 600 μS (1 mH)，2 ms (10 mH)

模拟量输出	SB 1223 AQ 1x12 位	
负载阻抗	电压： ≥ 1000 Ω	电流：≤ 600 Ω
RUN 到 STOP 时的行为	上一个值或替换值（默认值为 0）	
隔离（现场侧与逻辑侧）	无	
电缆长度（米）	100 米，屏蔽双绞线	
诊断：上溢/下溢		有
诊断：对地短路（仅限电压模式）		有
诊断：断路（仅限电流模式）		有

SB 1223 2x24VDC 2x24VDC



SB 1232 AQ 1



A.4 数字信号模块

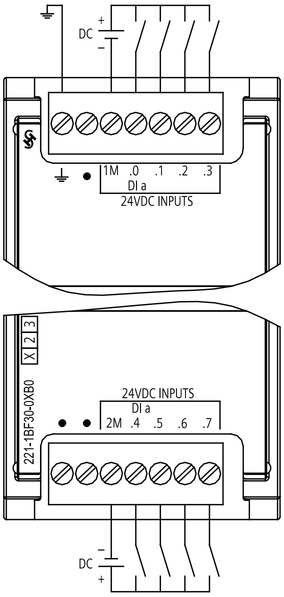
以下规范给出了一个适用于 S7-1200 的数字 SM 模块实例。更多相关信息，请参考 S7-1200 系统手册。

数字量输入信号模块示例 SM 1221

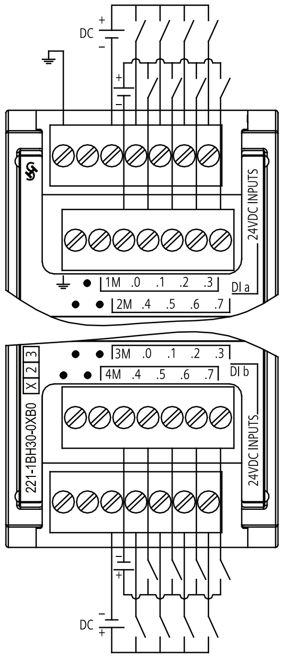
常规	SM 1221 DI 8x24VDC	SM 1221 DI 16x24VDC
输入点数 （可同时使用的输入点数）	8 (8)	16 (16)
尺寸 (W x H x D)	45 x 100 x 75 (mm)	45 x 100 x 75 (mm)
重量	170 g	210 g
功耗	1.5 W	2.5 W
电流消耗（SM 总线）	105 mA	130 mA
电流消耗 (24 VDC)	所用的每点输入 4 mA	所用的每点输入 4 mA

数字量输入	说明
输入类型	漏型/源型（IEC 1 类漏型）
额定电压	4 mA 时 24 VDC，额定值
允许的连续电压	最大 30 VDC
浪涌电压	35 VDC，持续 0.5 s
逻辑 1 信号（最小） 逻辑 0 信号（最大）	2.5 mA 时 15 VDC 1 mA 时 5 VDC
隔离（现场侧与逻辑侧） 隔离组	500 VAC，持续 1 min DI 8x24VDC： 2；DI 16x24VDC： 4
滤波时间 (ms)	0.2、0.4、0.8、1.6、3.2、6.4 和 12.8 （可选择，4 个为一组）
电缆长度（米）	500（屏蔽）；300（非屏蔽）

SM 1221 DI 8 x 24 VDC



SM 1221 DI 16 x 24 VDC



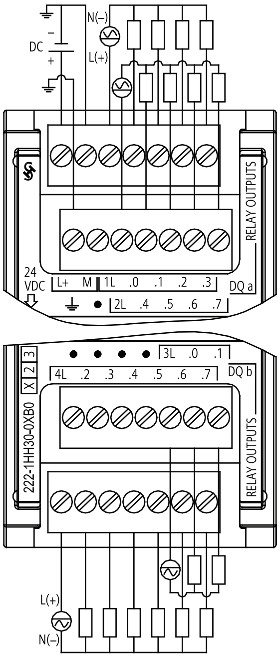
仅含输出的信号模块示例 SM 1222

常规	SM1222 DQ 16x 继电器	SM1222 DQ 16x24VDC
输出的点数和类型	16 点，继电器，干触点	16 点，固态 - MOSFET
尺寸 (W x H x D)	45 x 100 x 75 (mm)	45 x 100 x 75 (mm)
重量	260 g	220 g
功耗	8.5 W	2.5 W
电流消耗（SM 总线）	135 mA	140 mA
电流消耗 (24 VDC)	所用的每个继电器线圈 11 mA	不适用

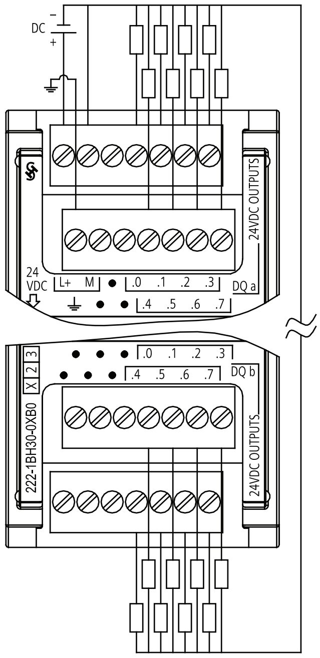
数字量输出	SM1222 DQ 16x 继电器	SM1222 DQ 16x24VDC
输出的点数和类型 （可同时使用的输出点数）	16 点，继电器，干触点 (16)	16 点，固态 - MOSFET (16)
电压范围	5 到 30 VDC 或 5 到 250 VAC	20.4 到 28.8 VDC

数字量输出	SM1222 DQ 16x 继电器	SM1222 DQ 16x24VDC
最大电流时的逻辑 1 信号 具有 10 KΩ 负载时的逻辑 0 信号	不适用	最小 20 VDC 最大 0.1 VDC
电流（最大）	2.0 A	0.5 A
灯负载	30 W DC/200 W AC	5 W
通态触点电阻	新设备最大为 0.2 Ω	最大 0.6 Ω
每点的漏泄电流	不适用	最大 10 μA
浪涌电流	触点闭合时为 7 A	8 A，最长持续 100 ms
过载保护	无	无
隔离（现场侧与逻辑侧）	线圈与触点：1500 VAC，持续 1 min 线圈与逻辑侧：无	500 VAC，持续 1 min
隔离电阻 断开触点间的绝缘	新设备最小为 100 MΩ 750 VAC，持续 1 min	不适用
隔离组	4	1
每个公共端的电流（最大）	10 A	8 A
电感钳位电压	不适用	L+ - 48 V，1 W 损耗
开关延迟	最长 10 ms	断开到接通最长为 50 μs 接通到断开最长为 200 μs
机械寿命（无负载） 额定负载下的触点寿命	10,000,000 个断开/闭合周期 100,000 个断开/闭合周期	不适用
RUN 到 STOP 时的行为	上一个值或替换值（默认值为 0）	上一个值或替换值（默认值为 0）
电缆长度（米）	500（屏蔽）；150（非屏蔽）	500（屏蔽）；150（非屏蔽）

SM 1222 DQ 16 x 继电器



SM 1222 DQ 16 x 24 VDC



数字量输入/输出组合的信号模块实例 SM 1223

常规	SM 1223 DI 16x24 VDC，DQ 16x 继电器	SM 1223 DI 16x24 VDC，DQ16x24 VDC
输入的点数和类型 （可同时使用的输入点数）	16 点，漏型/源型（IEC 1 类漏型） (16)	16 点，漏型/源型（IEC 1 类漏型） (16)
输出的点数和类型 （可同时使用的输出点数）	16 点，继电器，干触点 (16)	16 点，固态 - MOSFET (16)
尺寸 (W x H x D)	70 x 100 x 75 (mm)	70 x 100 x 75 (mm)
重量	350 g	310 g
功耗	10 W	4.5 W

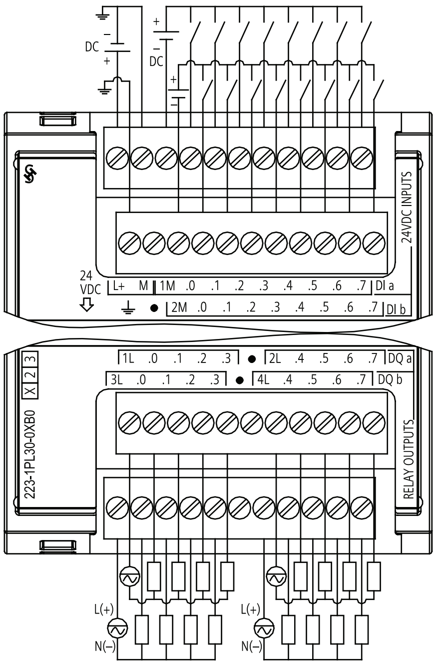
常规	SM 1223 DI 16x24 VDC，DQ 16x 继电器	SM 1223 DI 16x24 VDC，DQ16x24 VDC
电流消耗（SM 总线）	180 mA	185 mA
电流消耗 (24 VDC)	所用的每点输入 4 mA 所用的每个继电器线圈 11 mA	所用的每点输入 4 mA

数字量输入	说明
输入的点数和类型	16 点，漏型/源型（IEC 1 类漏型）
额定电压	4 mA 时 24 VDC，额定值
允许的连续电压	最大 30 VDC
浪涌电压	35 VDC，持续 0.5 s
逻辑 1 信号（最小） 逻辑 0 信号（最大）	2.5 mA 时 15 VDC 1 mA 时 5 VDC
隔离（现场侧与逻辑侧） 隔离组	500 VAC，持续 1 min 2
滤波时间 (ms)	0.2、0.4、0.8、1.6、3.2、6.4 和 12.8 ms（可选择，4 个为一组）
电缆长度（米）	500（屏蔽）；300（非屏蔽）

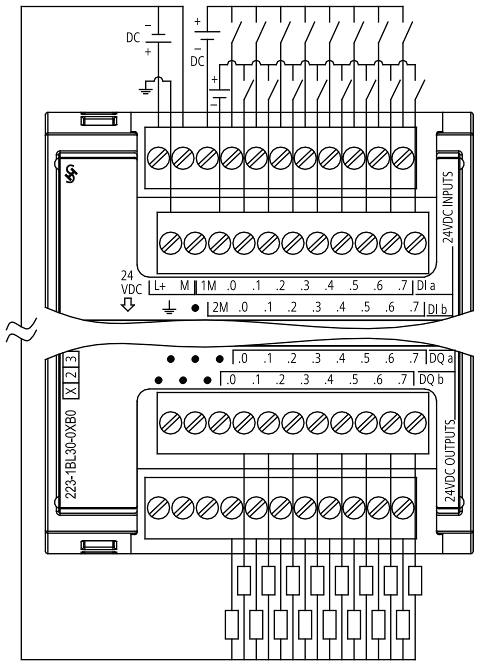
数字量输出	SM 1223 DI 16x24 VDC，DQ 16x 继电器	SM 1223 DI 16x24 VDC，DQ16x24 VDC
输出的点数和类型 （可同时使用的输出点数）	16 点，继电器，干触点 (16)	16 点，固态 - MOSFET (16)
电压范围	5 到 30 VDC 或 5 到 250 VAC	20.4 到 28.8 VDC
最大电流时的逻辑 1 信号 具有 10 KΩ 负载时的逻辑 0 信号	不适用	最小 20 VDC 最大 0.1 VDC
电流（最大）	2.0 A	0.5 A
灯负载	30 W DC/200 W AC	5 W
通态触点电阻	新设备最大为 0.2 Ω	最大 0.6 Ω

数字量输出	SM 1223 DI 16x24 VDC，DQ 16x 继电器	SM 1223 DI 16x24 VDC，DQ16x24 VDC
每点的漏泄电流	不适用	最大 10 μA
浪涌电流	触点闭合时为 7 A	8 A，最长持续 100 ms
过载保护	无	无
隔离（现场侧与逻辑侧）	线圈与触点： 1500 VAC，持续 1 min 线圈与逻辑侧： 无	500 VAC，持续 1 min
隔离电阻 断开触点间的绝缘	新设备最小为 100 MΩ 750 VAC，持续 1 min	不适用
隔离组	4	1
每个公共端的电流（最大）	8 A	8 A
电感钳位电压	不适用	L+ - 48 V，1 W 损耗
开关延迟	最长 10 ms	断开到接通最长为 50 μs 接通到断开最长为 200 μs
机械寿命（无负载） 额定负载下的触点寿命	10,000,000 个断开/闭合周期 100,000 个断开/闭合周期	不适用
RUN 到 STOP 时的行为	上一个值或替换值（默认值为 0）	
电缆长度（米）	500（屏蔽）；150（非屏蔽）	

SM1223 DI 16 x 24 VDC，DQ 16 x 继电器



SM 1223 DI 16 x 24 VDC，DQ 16 x 24 VDC



A.5 模拟信号模块

以下规范给出了一个适用于 S7-1200 的模拟 SM 模块实例。更多相关信息，请参考 S7-1200 系统手册。

常规	SM 1231 AI 4x13 位	SM 1234 AI 4x13 位 AQ 2x14 位	SM 1232 AQ 2x14 位
输入的路数和类型 （可选择，2 个为一组）	4 路，电压或电流 （差动）	4 路，电压或电流（差动）	0
输出的路数和类型	0	2 路，电压或电流	2 路，电压或电流
尺寸 W x H x D	45 x 100 x 75 (mm)	45 x 100 x 75 (mm)	45 x 100 x 75 (mm)
重量	180 g	220 g	180 g
功耗	1.5 W	2.0 W	1.5 W

常规	SM 1231 AI 4x13 位	SM 1234 AI 4x13 位 AQ 2x14 位	SM 1232 AQ 2x14 位
电流消耗（SM 总线）	80 mA	80 mA	80 mA
电流消耗 (24 VDC)	45 mA	60 mA（无负载）	45 mA（无负载）

模拟量输入	说明
输入类型	电压或电流（差动），可选择，2 个为一组
范围	±10 V、±5 V、±2.5 V 或 0 到 20 mA
满量程范围（数据字）	-27,648 到 27,648
过冲/下冲范围 （数据字）[1]	电压： 32,511 到 27,649/-27,649 到 -32,512 电流： 32,511 到 27,649/0 到 -4864
上溢/下溢（数据字）[1]	电压： 32,767 到 32,512/-32,513 到 -32,768 电流： 32,767 到 32,512/-4865 到 -32,768
精度	12 位 + 符号位
最大耐压/耐流	±35 V/±40 mA
滤波 [2]	无、弱、中或强
噪声抑制 [3]	400、60、50 或 10 Hz
阻抗	≥ 9 MΩ（电压）/250 Ω（电流）
隔离（现场侧与逻辑侧）	无
精度（25°C/0 到 55°C）	满量程的 ±0.1%/±0.2%
模数转换时间	625 µs（400 Hz 抑制）
共模抑制	40 dB，DC 到 60 Hz
工作信号范围	信号加共模电压必须小于 +12 V 且大于 -12 V
电缆长度（米）	100 米，屏蔽双绞线

[1] 请参考 S7-1200 系统手册，了解有关模拟量输入的电压和电流表示法。

[2] 请参考 S7-1200 系统手册，了解有关模拟量输入的阶跃响应时间。

[3] 请参考 S7-1200 系统手册，了解有关模拟量输入的采样速率。

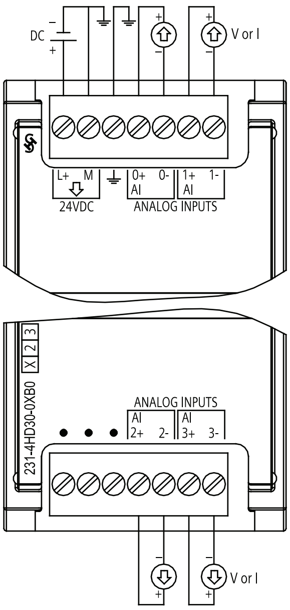
模拟量输出	说明
输出类型	电压或电流
范围	±10 V 或 0 到 20 mA
精度	电压： 14 位 电流： 13 位
满量程范围（数据字）[1]	电压： -27,648 到 27,648 电流： 0 到 27,648
精度（25°C/0 到 55°C）	满量程的 ±0.3%/±0.6%
稳定时间（新值的 95%）	电压： 300 µS (R)，750 µS (1 uF) 电流： 600 µS (1 mH)，2 ms (10 mH)
负载阻抗	电压： ≥ 1000 Ω 电流： ≤ 600 Ω
RUN 到 STOP 时的行为	上一个值或替换值（默认值为 0）
隔离（现场侧与逻辑侧）	无
电缆长度（米）	100 米，屏蔽双绞线

1 请参考 S7-1200 系统手册，了解有关模拟量输出的电压和电流表示法。

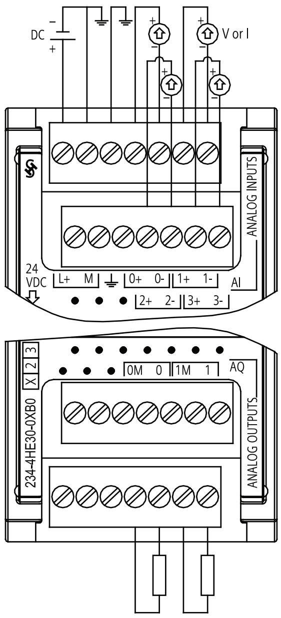
诊断	SM 1231 AI 4x13 位	SM 1234 AI 4x13 位 AQ 2x14 位	SM 1232 AQ 2x14 位
上溢/下溢	有 [1)]	有 [1)]	不适用
对地短路（仅限电压模式）	无	有（输出）	有
断路（仅限电流模式）	无	有（输出）	有
24 VDC 低压	有	有	有

1 如果对输入端施加大于 +30 VDC 或小于 -15 VDC 的电压，则结果值将是未知的，因此相应的上溢或下溢可能不会激活。

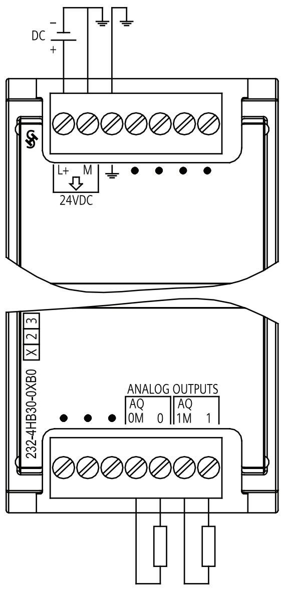
SM 1231 AI 4 x 13 Bit



SM 1234 AI 4x13bit AQ 2x14bit



SM 1232 AQ 2 x 14 bit

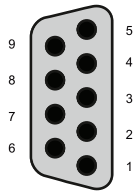


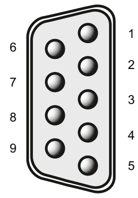
A.6 通信模块

以下规范给出了一个适用于 S7-1200 的 CM 模块实例。有关引脚信息或其它信息，请参考 S7-1200 系统手册。

常规	CM 1241 RS485	CM 1241 RS232
尺寸 (W x H x D)	30 x 100 x 75 (mm)	30 x 100 x 75 mm
重量	150 g	150 g
功率损失（损耗）	1.1 W	1.1 W
+5 VDC 电流	220 mA	220 mA

发送器和接收器		说明
发送器 (RS485)	共模电压范围	-7 V 到 +12 V，1 秒，3 VRMS 连续
	发送器差动输出电压	R_L = 100 Ω 时最小 2 V， R_L = 54 Ω 时最小 1.5 V
	终端和偏置	B 上 10K Ω 对 +5 V，PROFIBUS 针 3 A 上 10K Ω 对 GND，PROFIBUS 针 8
发送器 (RS232)	发送器输出电压	R_L = 3K Ω 时最小 +/- 5 V
	传送输出电压	最大 +/- 15 VDC
接收器	接收器输入阻抗	• RS485： 最小 5.4K Ω，包括终端 • RS232： 最小 3 K Ω
	接收器阈值/灵敏度	• RS485： 最低 +/- 0.2 V，典型滞后 60 mV • RS232： 最低 0.8 V，最高 2.4 V，典型滞后 0.5 V
	接收器输入电压 （仅限 RS232）	最大 +/- 30VDC
隔离	信号与外壳接地 信号与 CPU 逻辑公共端	500 VAC，1 分钟
电缆长度最大值，屏蔽电缆		• RS485： 1000 m • RS232： 10 m

CM 1241 RS485				
插针	说明	连接器（母）	插针	说明
1 GND	逻辑地或通信地		6 PWR	+5V 与 100 Ω 串联电阻： 输出
2	未连接		7	未连接
3 TxD+	信号 B (RxD/TxD+)： 输入/输出		8 TXD-	信号 A (RxD/TxD-)： 输入/输出
4 RTS	请求发送（TTL 电平）：输出		9	未连接
5 GND	逻辑地或通信地		SHELL	底盘接地

CM 1241 RS232				
插针	说明	连接器（公）	插针	说明
1 DCD	数据载波检测： 输入		6 DSR	数据设备就绪： 输入
2 RxD	从 DCE 接收数据： 输入		7 RTS	请求发送： 输出
3 TxD	传送数据到 DCE： 输出		8 CTS	允许发送： 输入
4 DTR	数据终端就绪： 输出		9 RI	振铃指示器（未用）
5 GND	逻辑地		SHELL	外壳接地

索引

A

C

D

F

H

T

子

存

安

定

密

工

帮

常

弹

强

循

打

扩

技

设

访

诊

调

路

输

连

通

配

重

长

门

防

项

额

高

北方区

北京
北京市朝阳区望京中环南路7号
邮政编码：100102
电话：(010) 6476 8888
传真：(010) 6476 4725

济南办事处
济南市舜耕路28号
舜华园商务会所5楼
邮政编码：250014
电话：(0531) 8266 6088
传真：(0531) 8266 0836

西安办事处
西安市高新区科技路33号
高新国际商务中心28层
邮政编码：710075
电话：(029) 8831 9898
传真：(029) 8833 8818

天津办事处
天津市和平区南京路189号
津汇广场写字楼1401室
邮政编码：300051
电话：(022) 8319 1666
传真：(022) 2332 8833

青岛办事处
青岛市香港中路76号，
青岛颐中皇冠假日酒店405室
邮政编码：266071
电话：(0532) 8573 5888
传真：(0532) 8576 9963

郑州
郑州市中原中路220号
裕达国贸中心写字楼2506室
邮政编码：450007
电话：(0371) 6771 9110
传真：(0371) 6771 9120

唐山办事处
唐山市建设北路99号
火炬大厦1308房间
邮政编码：063020
电话：(0315) 317 9450/51
传真：(0315) 317 9733

太原办事处
太原市府西街69号国际贸易
中心西塔16层1610B
邮政编码：030002
电话：(0351) 868 9048
传真：(0351) 868 9046

乌鲁木齐
乌鲁木齐市五一路160号
鸿福饭店C座918室
邮政编码：830000
电话：(0991) 582 1122
传真：(0991) 584 6288

洛阳办事处
洛阳市中州西路15号
洛阳牡丹大酒店4层415房间
邮政编码：471003
电话：(0379) 6468 0295
传真：(0379) 6468 0296

兰州办事处
兰州市东岗西路589号
锦江阳光酒店21层2111室
邮政编码：730000
电话：(0931) 888 5151
传真：(0931) 881 0707

烟台办事处
烟台市南大街9号
烟台金都大厦9层12室
邮政编码：264001
电话：(0535) 212 1880
传真：(0535) 212 1887

淄博办事处
淄博市张店区共青团西路95号
钻石商务大厦19层L单元
邮政编码：255036
电话：(0533) 230 9898
传真：(0533) 230 9944

银川
银川市北京东路123号
太阳神大酒店A区1507房间
邮政编码：750001
电话：(0951) 786 9866
传真：(0951) 786 9867

塘沽办事处
天津经济技术开发区第三大街
广场东路20号滨海金融街东区
E4C座三层15号
邮政编码：300457
电话：(022) 5981 0333
传真：(022) 5981 0335

石家庄
石家庄市中山东路303号
石家庄世贸广场酒店1309室
邮政编码：050011
电话：(0311) 8669 5100
传真：(0311) 8669 5300

东北区

沈阳分公司
沈阳市沈河区北站路59号
财富大厦E座13层
邮政编码：110013
电话：(024) 8251 8111
传真：(024) 2253 3626

大连办事处
大连市西岗区中山路147号
大连森茂大厦8楼
邮政编码：116001
电话：(0411) 8369 9760
传真：(0411) 8360 9468

哈尔滨办事处
哈尔滨市南岗区红军街15号
奥威斯发展大厦30层A座
邮政编码：150001
电话：(0451) 5300 9933
传真：(0451) 5300 9990

长春办事处
长春市西安大路569号
长春香格里拉大饭店401房间
邮政编码：130061
电话：(0431) 8898 1100
传真：(0431) 8898 1087

包头办事处
包头市钢铁大街66号国贸大厦
2107室
邮政编码：014010
电话：(0472) 590 8380
传真：(0472) 590 8385

鞍山
鞍山市铁东区东风街108号
鞍山东山宾馆2层
邮政编码：114010
电话：(0412) 558 1611
传真：(0412) 555 9611

呼和浩特办事处
呼和浩特市乌兰察布西路
内蒙古饭店15层1508房间
邮政编码：010010
电话：(0471) 693 8888 -1508
传真：(0471) 628 8269

华东区

上海分公司
上海市浦东新区浦东大道1号
中国船舶大厦10楼
邮政编码：200120
电话：(021) 3889 3889
传真：(021) 5879 3104

长沙办事处
长沙市五一大道456号，
亚大时代2101房
邮政编码：410011
电话：(0731) 446 7770
传真：(0731) 446 7771

南京办事处
南京市玄武区中山路228号
地铁大厦18层
邮政编码：210008
电话：(025) 8456 0550
传真：(025) 8451 1612

连云港办事处
连云港市连云区中华西路
千樱小区B幢3单元601室
邮政编码：222042
电话：(0518) 8231 3929
传真：(0518) 8231 3929

扬州
扬州市江阳中路43号
九州大厦7楼704房间
邮政编码：225009
电话：(0514) 8778 4218
传真：(0514) 8787 7115

杭州办事处
杭州市西湖区杭大路15号
嘉华国际商务中心1710室
邮政编码：310007
电话：(0571) 8765 2999
传真：(0571) 8765 2998

无锡
无锡市县前东街1号
金陵大饭店2401-2403室
邮政编码：214005
电话：(0510) 8273 6868
传真：(0510) 8276 8481

合肥办事处
合肥市濉溪路278号
财富广场27层2701、2702室
邮政编码：230041
电话：(0551) 568 1299
传真：(0551) 568 1256

宜昌分公司
宜昌市东山大道95号
清江大厦2011室
邮政编码：443000
电话：(0717) 631 9033
传真：(0717) 631 9034

徐州办事处
徐州市彭城路93号
泛亚大厦18层
邮政编码：221003
电话：(0516) 8370 8388
传真：(0516) 8370 8308

武汉分公司
武汉市汉口江汉区建设大道709号
建银大厦18楼
邮政编码：430015
电话：(027) 8548 6688
传真：(027) 8548 6668

温州办事处
温州市车站大道
高联大厦9楼B1室
邮政编码：325000
电话：(0577) 8606 7091
传真：(0577) 8606 7093

苏州办事处
苏州市新加坡工业园苏华路2号
国际大厦11层17-19单元
邮政编码：215021
电话：(0512) 6288 8191
传真：(0512) 6661 4898

宁波办事处
宁波市沧海路1926号
上东商务中心25楼2511室
邮政编码：315040
电话：(0574) 8785 5377
传真：(0574) 8787 0631

南昌办事处
南昌市北京西路88号
江信国际大厦1401室
邮政编码：330046
电话：(0791) 630 4866
传真：(0791) 630 4918

常州
常州市关河东路38号
九洲寰宇大厦911室
邮政编码：213001
电话：(0519) 8989 5801
传真：(0519) 8989 5802

绍兴办事处
绍兴市解放北路玛格丽特商业
中心西区2幢玛格丽特酒店10层
1020 室
邮政编码：312000
电话：(0575) 8820 1306
传真：(0575) 8820 1632/1759

扬中办事处
扬中市扬子中路199号
华康医药大厦703室
邮政编码：212200
电话：(0513) 8532 2488
传真：(0513) 8532 2058

华南区

广州分公司
广州市天河路208号
天河城侧粤海天河城大厦8-10层
邮政编码：510620
电话：(020) 3718 2888
传真：(020) 3718 2164

福州办事处
福州市五四路136号
中银大厦21层
邮政编码：350003
电话：(0591) 8750 0888
传真：(0591) 8750 0333

南宁办事处
南宁市 金湖路63号
金源现代城 9层 935室
邮政编码：530022
电话：(0771) 552 0700
传真：(0771) 552 0701

深圳分公司
深圳市华侨城汉唐大厦9楼
邮政编码：518053
电话：(0755) 2693 5188
传真：(0755) 2693 4245

东莞办事处
东莞市南城区宏远路1号
宏远大厦1403-1405室
邮政编码：523087
电话：(0769) 2240 9881
传真：(0769) 2242 2575

厦门办事处
厦门市厦禾路189号
银行中心21层2111-2112室
邮政编码：361003
电话：(0592) 268 5508
传真：(0592) 268 5505

佛山办事处
佛山市汾江南路38号
东建大厦19楼 K单元
邮政编码：528000
电话：(0757) 8232 6710
传真：(0757) 8232 6720

海口办事处
海口市大同路38号
海口国际商业大厦1042房间
邮政编码：570102
电话：(0898) 6678 8038
传真：(0898) 6678 2118

珠海
珠海市景山路193号
珠海石景山旅游中心229房间
邮政编码：519015
电话：(0756) 337 0869
传真：(0756) 332 4473

汕头
汕头市金海湾大酒店1502房
邮政编码：515041
电话：(0754) 848 1196
传真：(0754) 848 1195

湛江
湛江市经济开发区乐山大道31号
湛江皇冠假日酒店1616单元
邮政编码：524022
电话：(0759) 338 1616/3383232
传真：(0759) 338 6789

西南区

成都办事处
成都市人民南路二段18号
川信大厦18/17楼
邮政编码：610016
电话：(028) 8619 9499
传真：(028) 8619 9355

重庆办事处
重庆市渝中区邹容路68号
大都会商厦18层1809-12
邮政编码：400010
电话：(023) 6382 8919
传真：(023) 6370 0612

攀枝花
攀枝花市炳草岗新华街
泰隆国际商务大厦
B座16层B2-2
邮政编码：617000
电话：(0812) 335 9500 /01
传真：(0812) 335 9718

宜宾
宜宾市长江大道东段67号
华荣酒店0233号房
邮政编码：644002
电话：(0831) 233 8078
传真：(0831) 233 2680

绵阳
绵阳市高新区火炬广场
西街北段89号长虹大酒店
四楼商务会议中心
邮政编码：621000
电话：(0816) 241 0142
传真：(0816) 241 8950

西宁办事处
西宁市新宁路新宁花苑A座
紫恒国际公寓16楼21613室
邮政编码：800028
电话：(0971) 550 3390
传真：(0971) 550 3390

昆明办事处
昆明市 北京路155号
红塔大厦1204室
邮政编码：650011
电话：(0871) 315 8080
传真：(0871) 315 8093

贵阳办事处
贵州省贵阳市新华路
富中国际广场15层C座
邮政编码：550002
电话：(0851) 551 0310
传真：(0851) 551 3932

售后维修服务中心
西门子工厂自动化工程有限公司 (SFAE)
北京市朝阳区酒仙桥东路9号A1栋8层
邮政编码：100016
电话：(010) 8459 7000
传真：(010) 8459 7070

上海西门子工业自动化有限公司 (SIAS)
上海市中山南二路1089号
徐汇苑大厦22-25楼
邮政编码：200030
电话：(021) 5410 8666
传真：(021) 6457 9500

技术培训
北京：(010) 8459 7518
上海：(021) 6281 5933-305/307/309
广州：(020) 3810 2558
武汉：(027) 8548 6688-6400
沈阳：(024) 2294 9880/82518219
重庆：(023) 6382 8919-3002

技术资料
北京：(010) 6476 3726

技术支持与服务热线
电话: 400-810-4288
传真：(010) 6471 9991
E-mail: 4008104288.cn@siemens.com
Web:www. 4008104288.com.cn

亚太技术支持（英文服务）
及软件授权维修热线
电话：(010) 6475 7575
传真：(010) 6474 7474
E-mail:support.asia.automation@siemens.com

网址：www.ad.siemens.com.cn

西门子（中国）有限公司
工业业务领域
工业自动化与驱动技术集团

www.ad.siemens.com.cn